

# **Lujiteruiskubetonin kerrospaksuuden työnaikainen seuranta tunnelissa**

**Tarmo Röntilä**

**Sähkötekniikan korkeakoulu**

Diplomityö, joka on jätetty opinnäytteenä tarkastettavaksi  
diplomi-insinöörin tutkintoa varten Espoossa 20.11.2017.

**Työn valvoja:**

Prof. Arto Visala

**Työn ohjaaja:**

DI Ville Herva

Tekijä: Tarmo Rantila		
Työn nimi: Lujiteruiskubetonin kerrospaksuuden työnaikainen seuranta tunnelissa		
Päivämäärä: 20.11.2017	Kieli: Suomi	Sivumäärä: 7+52
Sähkötekniikan ja automaation laitos		
Professori: Automaatiotekniikka		Koodi: AS-84
Työn valvoja: Prof. Arto Visala		
Työn ohjaaja: DI Ville Herva		
<p>Tunnelirakentamisessa ruiskubetonointi on työvaihe, jossa tunnelin seinämiä tuetaan niille ruiskutettavalla betonilla. Nykyisin betonikerroksen paksuutta tarkkaillaan silmämääräisesti sekä harvahkoilla tuhoavilla näytteillä. Kattavamman ja ajantasaisen kerrospaksuustiedon avulla suunniteltu paksuus on helpompi saavuttaa ilman työmaan viivästymistä tai tarpeettoman suurta betoninkäyttöä.</p> <p>Tunnelityömaaolosuhteiden sekä ruiskubetonointiprosessin itsensä asettamista haasteista huolimatta instrumentointiratkaisuja kerrospaksuuden mittaamiseksi on useita. Näistä lupaavimmasta, ruiskubetonointilaitteistoon kiinnitetystä 3D-laserskannerista, toteutettiin menetelmän toimivuuden selvittämiseksi yksinkertaistettu prototyyppiversio.</p> <p>Prototyyppimittausjärjestelmän toiminnasta tehtyjen havaintojen perusteella instrumentointiratkaisu on toimiva, saadut tulokset hyödyllisiä, eikä järjestelmä häiritse ruiskubetonointityötä. Lisäksi viive skannaamisesta kerrospaksuuden visualisointiin todettiin ruiskubetonointiprosessin tarpeet huomioiden riittävän lyhyeksi. Joitain kysymyksiä järjestelmän luotettavuudesta jäi avoimeksi, minkä lisäksi esitettiin useampia jatkokehitysmahdollisuuksia järjestelmän eri yksityiskohtien suhteen. Puutteistaan huolimatta prototyyppijärjestelmää voi pitää onnistuneena ja 3D-laserskanneriin pohjautuvaa mittausjärjestelmää jatkokehityskelpoisena.</p>		
Avainsanat: ruiskubetonin kerrospaksuus, 3D-laserskannaaminen, pistepilvien kohdistaminen, reaaliaika		

Author: Tarmo Röntilä		
Title: Construction-time monitoring of supporting shotcrete lining thickness in tunnels		
Date: 20.11.2017	Language: Finnish	Number of pages: 7+52
Department of Electrical Engineering and Automation		
Professorship: Automation Technology		Code: AS-84
Supervisor: Prof. Arto Visala		
Advisor: M.Sc. (Tech.) Ville Herva		
<p>In tunnel construction, shotcreting is conducted to support the walls and roof of a tunnel with a concrete lining. Today the thickness of the lining is observed with crude visual inspection and sparse destructive testing. More comprehensive, close-to-real-time, information could be utilized to achieve the designed lining thickness both with less excess concrete and delays to the job site.</p> <p>Despite the challenges set by the environment of tunnel construction sites and shotcreting itself, multiple possible instrumentation solutions to measure the lining thickness exist. The most promising one was found to be a 3D laser scanner mounted on the shotcreting machinery. A prototype measurement system based on this solution was implemented to determine its applicability in a job site environment.</p> <p>Observing the prototype system it was concluded that the instrumentation solution is adequate, the acquired results are useful and the disturbance induced to the shotcreting process is minimal. Additionally, the delay from scanning to the visualization of the lining thickness was deemed to be close enough to real-time given the requirements of the process. Some questions on the robustness of the system remain unanswered and multiple details benefiting from further development were identified and possible solutions discussed. Regardless of its deficiencies the prototype system can be held successful and the 3D laser scanner based measurement system worthy of further development.</p>		
Keywords: shotcrete layer thickness, 3D laser scanning, point cloud registration, real-time		

## Esipuhe

Tämä työ tehtiin Viasys VDC Oy:n ja YIT Rakennus Oy:n Infrapalveluiden Kalliorakentaminen ja kaivokset -liiketoimintayksikön yhteistyönä. Havainto ruiskubetonikerroksen paksuuden mittaamisen kehittämismahdollisuuksista syntyi ja tarkentui useiden keskustelujen aikana jalostuen lopulta diplomityöaiheeksi.

Erityiskiitoksen mielenkiintoisen diplomityöaiheen löytymisestä ja mahdollistamisesta haluan Viasys VDC:n osalta antaa ohjaajalleni Ville Hervalle. Osansa näistä kiitoksista ansaitsevat myös YIT:n Ari Bergström ja Petri Raappana, joita haluan kiittää lisäksi minulle keskustellen syntyneestä tärkeästä perusymmärryksestä kalliorakentamisesta sekä ruiskubetonoinnista osana sitä.

Edellä mainitun lisäksi haluan kiittää YIT:tä työn käytännöllisen osan mahdollistamisesta niin laadukkaan laserskannerilaitteiston lainaamisen kuin skannattavien tunnelityömaidenkin osalta. Erityiskiitos myös niille YIT:n työntekijöille, jotka opastivat minua työmailla liikkuesssa sekä niiden käytännön toimintaan perehtymisessä.

Kollegoistani kiitän Mikko Rekolaa avusta prototyypin mittausohjelman kehityksen haasteiden ratkaisussa ja ajatustenvaihdosta työn aiheesta yleisemminkin. Tärkeää oli myös Saku Viita-ahon kiritys- kuin kirjoitustyönkin osalta. Kiitokset työn mahdollistavana voimana lisäksi valvojalleni professori Arto Visalalle; etenkin kärsivällisyydestä.

Lopuksi haluan vielä kiittää perhettäni tuesta diplomityöprosessin eri vaiheissa sekä sopivasta määrästä mielipiteitä ja kyseenalaistamista niin työn tavoitteiden, sisällön kuin aikataulunkin suhteen. Ilman kritiikkiä on välillä kovin vaikea nähdä omia käsiään pidemmälle.

Espoo, marraskuu 2017

Tarmo Rantilä



# Sisällysluettelo

<b>Tiivistelmä</b>	<b>ii</b>
<b>Tiivistelmä (englanniksi)</b>	<b>iii</b>
<b>Esipuhe</b>	<b>iv</b>
<b>Sisällysluettelo</b>	<b>v</b>
<b>Symbolit ja lyhenteet</b>	<b>vii</b>
<b>1 Johdanto</b>	<b>1</b>
<b>2 Ruiskubetonointi</b>	<b>2</b>
2.1 Ruiskubetoni materiaalina . . . . .	2
2.2 Merkitys ja prosessi . . . . .	2
2.3 Kerrospaksuuden seuranta . . . . .	3
<b>3 Mittausjärjestelmä</b>	<b>5</b>
3.1 Instrumentointimahdollisuudet . . . . .	5
3.2 Sensorityypit . . . . .	7
3.2.1 Rakenteellinen valo . . . . .	7
3.2.2 3D-laserskanneri . . . . .	8
3.2.3 Kulkuaiakamera . . . . .	8
3.2.4 Maatutka . . . . .	9
3.2.5 Takymetri . . . . .	9
3.3 Kokonaisratkaisu . . . . .	9
3.4 Rajattu prototyyppi . . . . .	12
<b>4 Tiedonkäsittely</b>	<b>13</b>
4.1 Tiedonkäsittelytarve . . . . .	13
4.2 Pistepilvet . . . . .	13
4.3 Kiihdytysrakenteet . . . . .	14
4.4 Esikäsittelymenetelmät . . . . .	15
4.4.1 Näytteistuksen harventaminen . . . . .	16
4.4.2 Poikkeavien havaintojen suotaminen . . . . .	17
4.4.3 Pintanormaalien laskenta . . . . .	18
4.5 Kohdistusmenetelmät . . . . .	19
4.6 Iteratiivinen lähimmän pisteen menetelmä . . . . .	21
4.6.1 Algoritmin yleiskuvaus ja askeleet . . . . .	21
4.6.2 Vastaavuuksien etsintä . . . . .	22
4.6.3 Vastaavuuksien painotus ja karsinta . . . . .	25
4.6.4 Virhometriikan minimointi ja konvergenssi . . . . .	27
4.7 Pistepilvien välinen erotus . . . . .	30

<b>5</b>	<b>Prototyyppijärjestelmä</b>	<b>32</b>
5.1	Laitteisto ja ohjelmistoalusta . . . . .	32
5.2	Mittausohjelmisto . . . . .	34
5.2.1	Skannien muuntaminen pistepilviksi . . . . .	34
5.2.2	Esikäsittely ja kiihdytysrakenteet . . . . .	35
5.2.3	Skannien kohdistaminen . . . . .	36
5.2.4	Betonikerroksen paksuuden määrittäminen ja visualisointi . . . . .	37
5.3	Havainnot ja toimivuus . . . . .	37
5.3.1	Skannien kohdistuminen . . . . .	37
5.3.2	Betonikerroksen paksuus . . . . .	40
5.3.3	Käyttökelpoisuus työmaaoloissa . . . . .	42
5.4	Jatkokehitysmahdollisuudet . . . . .	43
<b>6</b>	<b>Johtopäätökset</b>	<b>46</b>
	<b>Viitteet</b>	<b>48</b>

# Symbolit ja lyhenteet

## Notaatio

$\mathbf{A}$	Matriisit isoin lihavoiduin symbolein
$\mathbf{b}$	Kolumnivektorit pienin lihavoiduin symbolein
$\mathbf{a}_i$	Matriisin $\mathbf{A}$ järjestyksessä $i$ :s kolumnivektori
$E, d$	Skalaarit lihavoimattomin symbolein
$\mu_{\mathbf{A}}$	Matriisin $\mathbf{A}$ kolumnivektorien keskiarvovektori
$\ \mathbf{d}\ $	Vektorin pituus eli euklidinen normi
$\angle(\mathbf{a}, \mathbf{d})$	Vektorien välinen kulma
$\mathcal{O}(n)$	Algoritmin asymptoottinen kompleksisuus $n$ :n suhteen

## Symbolit

$\mathbf{P}$	Pistepilvi, mittauspistepilvi
$\mathbf{Q}$	Referenssipistepilvi
$\mathbf{p}$	Pistepilven piste
$\mathbf{q}$	Referenssipistepilven piste
$\mathbf{n}$	Pistepilven pisteelle laskettu normaali
$\mathbf{e}$	Mittaus- ja referenssipilvien pistevastaavuuksien erotus
$\mathbf{T}$	Mittaus- ja referenssipilven kohdistava transformaatio
$\mathbf{R}$	$\mathbf{T}$ :n rotaatio-osa
$\mathbf{t}$	$\mathbf{T}$ :n translaatio-osa

## Lyhenteet

lidar	LIght Detection And Ranging, optinen tutka, laserskanneri
GPR	Ground Penetrating Radar, maatutka
IMU	Inertial Measurement Unit, inertiamittari
PCL	Point Cloud Library, ohjelmakirjasto erityisesti laserskannerityylisten pistepilvien käsittelyyn
PCA	Prime Component Analysis, pääkomponenttianalyysi, menetelmä datajoukon ulotteisuuden pienentämiseksi
ICP	Iterative Closest Point, iteratiivinen pistepilvien hienokohdistusmenetelmä
NDT	Normal Distribution Transform, vaihtoehtoinen ICP:tä uudempi hienokohdistusmenetelmä
SVD	Singular-Value Decomposition, pääakselihajotelma
SDK	Software Development Kit, sovelluskehityspaketti
HSV	Hue, Saturation, Value, väriavaruus

# 1 Johdanto

Tunnelit ovat erottamaton osa modernia infrastruktuuria. Maanalaisena rakenteen ne vaativat erillistä tuentaa romahdusvaaran ehkäisemiseksi sekä työma- että käyttövaiheissa. Eräs tärkeä menetelmä tunneleiden seinämien tukemiseksi on ruiskubetonointi eli niille paineella ruiskutettu betonikerros. Lujitusvaikutuksen takaa betonin laadun lisäksi sopiva kerrospaksuus [1].

Nykyisin ruiskubetonikerroksen paksuutta seurataan silmämääräisesti sekä harvahkoilla pistemäisillä näytteillä [2], [3]. Arvioita voi esittää myös kuluneen betonin määrän ja ruiskutetun pinta-alan pohjalta [2]. Tunnelin pinnan vaihtelevat muodot ja kimpoamahukka tekevät menetelmistä vääjäämättä suhteellisen epävarmoja etenkin kattavuusmielessä. Betonikerroksen oikea paksuus mitattujen pisteiden välillä on pitkälti ruiskuttajan ammattitaidon varassa [1], [4]. Pätevinkään ruiskuttaja ei kuitenkaan kykene aistinvaraisesti arvioimaan betonikerroksen paksuutta luotettavasti koko ruiskutetulta alueelta.

Kattavan kerrospaksuustiedon puute on ongelmallista, sillä jälkikäteen havaittujen puutteiden korjaaminen on hankalaa ja ”varmuuden vuoksi” ruiskuttaminen hukkaa sekä aikaa että betonia [2], [3], [5]. Eräs osaratkaisu tarkemmin suunnitelman mukaisen kerroksen saavuttamiseksi on ruiskutuksen automatisointi. Sekä Girmscheid ja Moser [6] että Nabulsi, Rodriguez ja Rio [5] ovat tutkineet tätä suhteellisen hyvin tuloksin helpoissa ruiskutusolosuhteissa. Hankalampien pintojen ruiskutuksessa tarkka, kattava ja nopea tieto betonikerroksen paksuudesta lienee ainoa keino parantaa lopputuloksen luotettavuutta. Kattavamman ja tarkan tiedon avulla voidaan varmistaa betonikerroksen riittävyys. Nopea, työnaikainen, saatavuus taas mahdollistaa optimistisemmän ruiskutustavan niin ihmiselle kuin robotillekin pienten puutteiden välittömän korjaamisen ollessa helppoa.

Tämän työn tarkoituksena on selvittää mahdollisia menetelmiä ruiskubetonikerroksen paksuuden työnaikaiseen eli riittävän reaaliaikaiseen seurantaan, valita näistä käyttökelpoisin ja ennen kaikkea tuottaa prototyyppijärjestelmä, jonka pohjalta arvioida tekniikan käyttökelpoisuutta ja mahdollisia rajoitteita tunnelityömaaolosuhteissa. Tätä silmällä pitäen työn toisessa luvussa esitellään ruiskubetoni aineena sekä ruiskubetonointi prosessina sen instrumentoinnin kannalta merkittävällä tarkkuudella. Lisäksi eritellään tarkemmin betonikerroksen paksuuden seurannan nykytila ja parannustarve. Kolmannessa luvussa käydään läpi joitain instrumentointiratkaisuja, esitetään näistä parhaaksi katsottu osana mahdollista mittausjärjestelmää sekä rajataan työn puitteissa toteutettava prototyyppijärjestelmä. Neljännessä luvussa esitetään ne tietojenkäsittelymenetelmät, joiden avulla mittausjärjestelmä selvittää kerätyn sensoridatan perusteella betonikerroksen paksuuden. Viidennessä luvussa taas esitellään toteutetun prototyyppijärjestelmän laitteisto ja ohjelmisto sekä sen toiminnasta tehdyt havainnot ja jotain tunnistettuja jatkokehitysmahdollisuuksia. Lopuksi tehdään yhteenveto löydöksistä ja esitetään johtopäätöksiä prototyypillä koestetun mittausjärjestelmäkokonaisuuden toimivuudesta ruiskubetonin kerrospaksuuden työnaikaisesta seurannasta tunnelityömaaolosuhteissa.

## 2 Ruiskubetonointi

Ruiskubetonointi on tärkeä osa tunneliseinämän lujittamista niin tunnelityöturvallisuuden kuin tunnelin lopullisen käytönkin kannalta. Sen rooli on pitää seinämien irtoamisherkkä aines paikallaan estäen spontaanin materiaalin tippumisen tunnelin seinämiltä ja vakauttaen tunnelia ympäröivää kallioperää [7], [8]. Tässä luvussa kuvataan ruiskubetoni materiaalina sekä ruiskubetonointiprosessi osana tunnelityömaan kokonaisprosessia relevanteilta osin. Lisäksi esitellään nykyisin käytetyt ruiskubetonin kerrospaksuuden seurantamenetelmät ja niiden suurimmat puutteet.

### 2.1 Ruiskubetoni materiaalina

Ruiskubetoni on pohjimmiltaan sementti-vesi-seos, joka ruiskutetaan suurehköllä nopeudella kohdealueelle paineilman avulla. Sen suurin ero perinteiseen valettuun betoniin on nimenomaan käsittelytapa, vaikka koostumuserojakin on [9]. Merkittävimmät edut valubetoniin nähden ovat ruiskutettavan betonimassan nopeuden aiheuttama tiivistyminen ja vähäisempi tarve muotti- ja tukirakenteille sekä mahdollisuus tuottaa vaihtelevan paksuisia ja muotoisia pintoja [1], [9].

Ruiskubetonointia tehdään kahdella eri tavalla; niin sanotuilla kuiva- ja märkämenetelmillä [1]. Kuivamenetelmässä vesi ja sementti sekä muut ainesosat sekoittuvat keskenään niin myöhään kuin mahdollista eli ruiskun suulla. Märkämenetelmässä taas vesi ja sementti sekä mahdolliset muut aineet ovat suurimmaksi osaksi jo seostettuja ennen pumppauslaitteiston läpi kulkemista. Kuivamenetelmän hyviin puoliin kuuluu suuri joustavuus pinnanmuotojen ja betoniseoksen laadun suhteen. Märkämenetelmä taas mahdollistaa tasaisemman betonilaadun ja suuremman applikaatiokapasiteetin sekä pienemmän kimpoamahukan [4], [9]. Tunneliolosuhteissa käytetään usein märkämenetelmää, sillä ruiskutettavaa pintaa on paljon ja betonin laatu on turvallisuuden kannalta kriittistä siinä missä paksuuskin.

Menetelmästä riippumatta ruiskubetoniin lisätään monesti sen ominaisuuksia parantavia aineita [1]. Näistä yleisimpiä ovat etenkin kattoon ruiskutettaessa käytännössä välttämätön kiihdytin, joka kuivattaa ja vahvistaa betonin nopeammin vähentäen kimpoamahukkaa. Toinen merkittävä ruiskubetonin ominaisuuksia parantava tekijä on siihen sekoitettava kuituaines; sekä metallia, lasia että muovia käytetään [1]. Näistä yleisin on nykyään metallikuitu, mutta tyypistä riippumatta kuidun käyttö vähentää betonin halkeamisherkkyyttä niin dynaamisessa kuin staattisessakin rasituksessa. Kuidun ansiosta betoni siis kestää paremmin sekä kuivamiskutistumista että tunnelin laajennustöiden aiheuttamaa tärinää.

### 2.2 Merkitys ja prosessi

Ruiskubetonointi on keskeinen osa seinämänlujitusta etenkin uudessa itävaltalaisessa tunnelinrakennustavassa (engl. New Austrian Tunneling Method) ja sen johdannaisissa, joissa se suoritetaan mahdollisimman pian louhinnan jälkeen paljastuneen kalliopinnan vakauttamiseksi [8]. Tuetaan tarvittavan kokonaisbetonikerroksen paksuus vaihtelee, mutta on karkeasti senttimetrien tai kymmenien suuruusluokassa.

Seinämän tarvitessa lisätuentaa se suoritetaan ensisijaisesti metallisin pultein ja verkoin [7], [8]. Niidenkin päälle kuitenkin voidaan kallion laadun niin vaatiessa ruiskuttaa lisää betonia, jolloin muodostuva rakenne vastaa verkkojen tapauksessa monilta osin teräsbetonia.

Ruiskubetonointityö voidaan suorittaa joko siten, että ruiskuttaja käsittelee suoraan ruiskutusletkua ja -suutinta tai nykyisin useimmiten liikkuvaan alustaan kiinnitettyä robottikäsivartta kauko-ohjaten. Robottikäsivarren hyviin puoliin kuuluu sen kyky kannatella painavampaa letkua sekä ruiskun suuttimen pyörivän liikkeen automaatio. Näistä ensimmäinen tehostaa betonointia mahdollistamalla suurempien kuutiomäärien ruiskutuksen lyhyemmässä ajassa ja jälkimmäinen pienentää kimpoamahan määrää [4]. Kauko-ohjatun suuttimen käyttäminen myös lisää ruiskuttajan työturvallisuutta poistamalla tarpeen tehdä työtä vastaruiskutetun betonikerroksen välittömässä läheisyydessä. Tuoreesta betonikerroksesta voi irrota ja tippua isoja paloja esimerkiksi liiallisen kerrospaksuuden tai huonon laadun takia erityisesti ruiskutuksen aikana, mutta mahdollisesti vielä tunteja sen jälkeenkin [2].

Aiemmin mainittujen tekijöiden lisäksi kimpoamahukkaa pyritään pienentämään betonoimalla pintaa kattoa kohti edeten [4]. Tämä etenemisjärjestys kompensoi ruiskubetonin heikompaa tarttuvuutta vaakasuorempien tasojen alapinnoille sen hyvällä tarttuvuudella tuoreeseen betoniin. Tästäkin huolimatta kimpoamahukka on yleisesti kohtuullisen runsaat 10–25 % ruiskutettavasta pinnasta riippuen [4], [10]. Ruiskubetonin erittäin nopean kuivumisen ja kovettumisen johdosta pois kimpoava aines on raekooltaan ja -massaltaan soramaista.

Yleisesti betonia ruiskutetaan tunnelissa joitain metrejä leveä ja useampia metrejä korkea ala kerrallaan ja tavanomaisesti yhdellä ruiskutuksella pyritään korkeintaan 15 cm kerrospaksuuteen [9]. Paksumpi kokonaiskerros tuotetaan ruiskuttamalla uusia kerroksia kuivuneiden päälle. Yhdellä kertaa ruiskutettavan alueen kokoa rajoittaa periaatteessa vain ruiskutuslaitteiston ylettyvyys, joka voi isommilla robottikäsivarsilla olla kymmenen metriä tai enemmänkin [4]. Tunnelissa käytännön rajoitteena toimii lisäksi jossain määrin betonin saatavuus; yleisimmin käytetyn märkämenetelmän betoni toimitetaan useimmiten ruiskubetonointipaikalle kuorma-autoilla, joita ei välttämättä mahdu kulkemaan tunnelissa useampaa rinnakkain. Kaiken kaikkiaan ruiskutustyössä on vaihtelevan mittaisia taukoja ruiskutuslaitteiston siirtyessä ja betonikuorman vaihtuessa. Yleisimmin kuorman vaihtuminen kestää kymmenisen minuuttia ja laitteiston siirtäminen lyhimmillään vain muutaman minuutin.

## 2.3 Kerrospaksuuden seuranta

Ruiskubetonikerroksen paksuutta valvotaan nykyään silmämääräisesti sekä hajanaisesti suoritetuilla tuhoavilla testeillä. Tuoreehkon betonin paksuus voidaan määrittää siihen lyötävällä terävällä esineellä, kuivuneeseen tai laadultaan erityisen kovaan betoniin tarvitaan poraa [3]. Toinen, huomattavasti työläämpi, vaihtoehto on kallio-pintaan asennettavat metalliset tapit, joiden ulkonema asetetaan vastaamaan tavoiteltua betonikerroksen paksuutta [3]. Paksuutta valvotaan paitsi työmaavaiheessa sekä ruiskuttajan että työnjohdon toimesta, myös jälkikäteen työmaan tunnelitöiden valmistuttua dokumentointi- ja laadunvarmistustarkoituksessa [2], [3].

Sekä työmaavaiheessa että laadunvarmistustarkoituksessa tapahtuvien mittausten suurin ongelma on erittäin huono kattavuus; mittaustulokset ovat kaikilla nykymentelmillä erittäin pistemäisiä. Kalliopinnan muoto betonikerroksen alla kuitenkin vaihtelee, jolloin betonikerroksen paksuus voi mittausten välillä vaihdella merkittävästi. Työmaavaiheessa lisäongelmana on mittausten tekeminen ihmisen ylettyvyyttä korkeammalta eli esimerkiksi kimpoamahukan vuoksi tärkeästä katosta [3].

Kerrospaksuustiedon heikko kattavuus on ongelmallista, sillä se asettaa varmuuden betonikerroksen suunnitelmanmukaisuudesta jossain määrin kyseenalaiseksi. Toinen, tästä seuraava, merkittävä ongelma on väistämätön tarve korjata betonikerrosta jälkikäteen tehtyjen havaintojen johdosta [2]. Työnjohdon tarkastaessa betonikerroksen paksuutta esimerkiksi ruiskutustyötä seuraavana päivänä vahinko on vielä pieni, sillä testattu ala voidaan pienellä aikatauluhaitalla yksinkertaisesti ruiskuttaa tarpeellisilta osin uudestaan. Jälkikäteen esimerkiksi asiakasta tai viranomaista varten tehdyissä laadunvarmistusmittauksissa löytyneiden puutteiden korjaaminen on hankalampaa ja kalliimpaa. Tällöin tunnelissa on yleisesti käynnissä jo muita, korjausruiskutuksesta häiriintyviä, rakennustöitä, eikä työmaalla välttämättä enää ole ruiskutuskalustoa tai pätevää ruiskuttajaakaan [3].

Kerrospaksuuden riittävyyden ristiriitainen käänköpuoli on sen minimaalisuus. Kaikki suunniteltua paksumpaan betonikerrokseen johtava ruiskutus hukkaa sekä betonia että aikaa, jotka kummatkin tuottavat lisäkuluja [2]. Käytännössä betonikerroksen kattavan minimaalisuuden saavuttaminen toteutuu nykyään vielä riittävyystäkin heikommin, sillä riittävyys on näistä kriittisempi tekijä. Lisäksi minimaalisuuteen voi vaikuttaa vain ruiskuttaja itse työtä tehdessään, eikä kerrosta ole lähtökohtaisesti järkeä jälkikäteen ohentaa.

Kaiken kaikkiaan kattavampi tieto betonikerroksen paksuudesta lisäisi luottamusta kerrospaksuuden suunnitelmanmukaisuuteen ja ehkäisisi jälkikäteen suoritettavia korjaustöitä. Riittävän nopeasti saatu kattava tieto taas hyödyttäisi ruiskutettavan betonin määrän minimoinnissa ja nopeuttaisi sekä helpottaisi virheiden korjaamista entisestään. Mikäli betonikerroksen paksuus saadaan mitattua ja visualisoitua ruiskutustyön lyhyen tauon aikana, ruiskuttaja voi arvioida syntyneitä kerrospaksuutta optimistisemmin ja korjata ohueksi jääneitä alueita heti työn jatkuessa. Nopea palautte voi lisäksi auttaa ruiskuttajaa kehittämään omaa osaamistaan kerrospaksuuden arvioinnissa [3].

### 3 Mittausjärjestelmä

Mahdollisia instrumentointiratkaisuja ruiskubetonikerroksen paksuuden määrittämiseksi on useita. Tämän työn puitteissa merkittävimmät niissä huomioitavat tekijät ovat tarve mahdollisimman suurelle kattavuudelle sekä nopeasti muodostettaville mittaustuloksille. Myös määritet ja pimeät tunneliolosuhteet sekä ruiskubetonointiprosessin itsensä asettamat rajoitteet on syytä huomioida. Tässä luvussa käsitellään mahdollisia instrumentointiratkaisuja ruiskubetonikerroksen paksuuden mittaamiseksi tuotetun datan käyttökelpoisuuden lisäksi nimenomaan toimintaympäristön asettamien rajoitteiden kannalta. Lisäksi esitetään perusteltu mittausjärjestelmän kokonaisratkaisu sekä rajataan tämän työn puitteissa toteutettava prototyyppi, jonka avulla ratkaisun toimivuutta on mahdollista arvioida.

#### 3.1 Instrumentointimahdollisuudet

Kuten muidenkin aktiivisten työmaiden, on tunnelityön instrumentoinnin tärkein rajoittava tekijä se, ettei työtä voi olennaisesti hankaloittaa ja hidastaa. Täyteen häiriöttömyyteen ei toisaalta ole välttämätöntä pyrkiä, sillä instrumentoinnilla tietysti tavoitellaan hyötyjä, jotka voivat olla suurempiakin kuin aiheutuneet haitat. Kuitenkin tunneli ympäristönä asettaa keskimääräistä kovemmat vaatimukset häiriöttömyydelle. Tehokasta ja tunneliprojektin aikataulullista kriittistä polkua edistävää työtä pystytään usein tekemään vain niissä tunnelin osissa, joista sitä kullakin hetkellä laajennetaan.

Karkeasti ottaen prosessia seuraavia sensoreita on mahdollista sijoittaa jalustalle maahan, ruiskutusalueeseen itseensä, ruiskutusalueen robottikäsiä varten, ruiskutus-suuttimen läheisyyteen tai mahdollisesti osaksi ruiskuttajan laitteistoa esimerkiksi kypärään tai kokonaan erillisenä laitteena. Lisäksi useamman sensoriyksikön tai -tyypin käyttäminen on vakavasti otettava vaihtoehto ympäristön aiheuttamien ongelmien minimoimiseksi ja luotettavuuden maksimoimiseksi.

Jalustalla maassa seisovan sensorin etuja ovat vapaa siirrettävyys eli mahdollisuus vaihtaa havainnointipaikkaa tarpeen mukaan. Lisäksi esimerkiksi takymetrejä varten käytetään jo nykyisinkin kolmijalkajalustoja, joten ratkaisun toimintavarmuus on jalustojen saatavuuden osalta hyvä. Ratkaisu on myös havainnoinnin itsensä kannalta helppo, sillä sensori on asemoitavissa ja pysyy näin paikallaan sekä yhden mittauksen aikana että useamman välillä. Toisaalta jalustalla seisova sensori on suurimmassa vaarassa rikkoutua kaatumalla; etenkin tuoreen tunnelin pohja ei ole välttämättä kovin tasainen. Samoin ruiskutusalueesta ja robottikäsiä kuin myös ruiskuttaja ja mahdollinen avustaja liikkuvat kaikki kohtuullisen pienellä alueella, johon myös sensorin pitäisi mahtua ilman merkittäviä katveita (ks. kuva 1 seuraavalla sivulla). Tarve ylipäänsä siirrellä sensoria käsin on lisävaiva ja hidaste. Myös tiedonsiirto sensorilta laskentayksikölle voi olla hankalaa joko langattoman siirtokaistan riittävyyden tai langallisen ratkaisun mukanaan tuomien lisäkaapeleiden takia.

Sensorin sijoittaminen ruiskuttajaan itseensä tarjoaa kohtuullisen katveettoman näkökulman ruiskuttajan työskennellessä seisoen. Lisäksi sensorin läheisyys ruiskuttajan eli mittaustiedon ensisijaisen käyttäjän kanssa pienentää tarvetta pitkille ja



hankalille johdotuksille. Ruiskuttajan mukana sensori myös kulkee melko vaivattomasti. Täsmällisestä sijoituksesta ja käyttötavasta riippuen, esimerkiksi kypärässä tai käsikäyttöisesti, käyttäminen voi vaatia jonkin verran vaivannäköä. Tässä lähestymistavassa datan prosessointia hankaloittaa ruiskuttajan liike, sillä sensori ei pysy mittausten välillä tai välttämättä niiden aikanakaan paikallaan. Sensori ei myöskään voi olla kovin suuri tai painava, jotta se ei häiritse ruiskuttajan työtä – sama koskee myös sensorin suojausta, joka on ruiskuttajan päälle väistämättä roiskuvan betonin takia välttämättömyys. Lisäksi joitain ruiskutuslaitteistoja operoidaan hytistä, josta ei ole välttämättä järkevää näkyvyyttä ympäristöön, vaikka ruiskuttaja itse ruiskutettavan pinnan näkee.

Ruiskutusalan osiin liitettävissä sensoriratkaisuissa kaikissa hyvä on, että ne kulkevat mukana kätevästi ja tarjoavat kohtuullisen hyvät edellytykset suoja- ta sensorilaite, esimerkiksi jonkinlaisella nostettavalla suojakuvulla. Tämä ratkaisu mahdollistaa myös raskaamman tietokonelaitteiston käyttämisen ilman työtä hanka-



Kuva 1: Ruiskubetonointialusta, sen robottikäsivarsi ja ympäröivä tila.

loittavia johdotuksia tai langattoman verkon ongelmia. Kuten ruiskuttajaankin, myös ruiskutuslaitteistoon kiinnitetty sensorilaitteisto todella tarvitsee suojausta. Ruiskun suun läheisyyteen kiinnitettäessä seinän läheisyys aiheuttaa samankaltaisen tilanteen kuin kypäräkiinnitys; sensori liikkuu jatkuvasti, eikä ylipäättään ole aina samassa orientaatioissa. Sama koskee myös lievemmin myös robottikäsiä. Toisaalta näillä tavoin pääsee luontevasti mahdollisimman lähelle seinämää ja katveen aiheuttamat ongelmat datan hankinnassa ovat minimaaliset. Etäisyydeksi seinään jää tällöin noin puoli metriä. Kiinnitys alustaan ei ole aivan yhtä katveeton, vaikka silti tältäkin osin käyttökelpoinen, vaihtoehto.

## 3.2 Sensorityypit

Käyttötapausten sensoritekniikkaa rajoittaa sen sijoitusrajoitukset kuten massa ja tilavuus. Suotavia piirteitä ovat lisäksi runsaan datamäärän tuottaminen kohtuullisen nopeasti sekä mahdollisuus suorittaa mittaukset joidenkin metrien etäisyydeltä, jolloin sensoria ei tarvitse välttämättä viedä aivan vastaruiskutetun betonikerroksen viereen. Käytännössä näihin piirteisiin sopivat parhaiten erilaiset 3D-sensorit. Seuraavaksi esitellään käyttötapausten kannalta käyttökelpoisimmaksi arvioidut sensoritekniikat työn tavoitteen vaatimalla tarkkuudella.

### 3.2.1 Rakenteellinen valo

Rakenteellinen valo (engl. structured light) on kolmiomittaukseen perustuva tapa luoda syvyys- eli “3D”-kuva. Kolmiomittauksen vaatima tieto järjestetään heijastamalla yhden tai useamman aktiivisen komponentin eli projektorin avulla tunnettu kuvio tarkkailtavaan ympäristöön, jota yksi tai useampi passiivinen kamerakomponentti tarkkailee. Kun komponenttien orientaatiot suhteessa toisiinsa tiedetään, voidaan kuvion havaitusta vääristymästä mitata kunkin pisteen etäisyys laitteistosta. Järjestely siis vastaa toimintaperiaatteeltaan stereokamerajärjestelmää, jossa projektorit toimivat monessa mielessä käänteisenä kamerana.

Useimmiten rakenteellista valoa käytettäessä sensorilaitteessa on yksi valoprojektori ja kaksi kameraa sen molemmin puolin. Tällöin välttyään katveilta sensorin näkökentän keskiosassa. Saadun syvyystiedon hyvyyteen vaikuttaa olennaisesti käytetyn aktiivisen valon kuvio sekä muut ominaisuudet ja aiheesta onkin tehty runsaasti tutkimusta (ks. [11]). Tärkeimmät käytetyt kuviot lienevät yksinkertainen ja joustava ristikko sekä satunnaiskuviot, joiden hyviin puoliin kuuluu esimerkiksi käsittelyn helpous ja siten tehokkuus [12]. Lisäksi käytetyn “valon” allonpituus voi olla näkyvän valon alueen ulkopuolinen, jolloin saadaan tarvittaessa kerättyä myös vääristymätön RGB-informaatio. Infrapunataajuudet ovat suosittu vaihtoehto [13].

Rakenteellisen valon hyviin puoliin kuuluvat hyvä syvyystarkkuus ja joillekin menetelmille myös kyky tuottaa kokonainen syvyyskuva yhdellä valotuksella. Myös menetelmän syvyysresoluutio on hallituissa olosuhteissa lähietäisyydellä loistava [14]. Toisaalta menetelmän tarkkuus ja käyttökelpoisuus ylipäättään kärsii huomattavasti kun etäisyys sensorista kasvaa tai mitattava pinta on epäsäännöllinen [15], [16]. Tekniikan hyväksi puoleksi voi laskea myös markkinoille viime vuosina tulleet sitä

käyttävät useat kompaktit sensorilaitteistot, kuten Microsoft Kinect (XBox 360-versio), PrimeSense ja Structure.io, jotka ovat myös hintansa puolesta erittäin houkuttelevia. Niiden tarjoaman syvyyskuvan pikseliresoluutio on kuitenkin heikohko ja ilmoitettu maksimihavainnointietäisyys 5 m luokassa, mikä on rakenteelliselle valolle kohtuullista, mutta tämän työn käyttötapauksessa ongelmallista.

### 3.2.2 3D-laserskanneri

Useasti tuttavallisemmin laserskannaukseksi kutsuttu lidar-tekniikka (engl. light detection and ranging) lienee perinteisin ja periaatteeltaan yksinkertaisin tapa tuottaa syvyyskuvaluonteista tietoa. Sensorilaitte lähettää lasersäteen ja odottaa sen kimpoavan takaisin – kun tähän kulunut aika tai moduloidun säteen vaihemuutos kyetään mittaamaan, saadaan valon nopeuden avulla laskettua etäisyys mittapisteeseen [16]. Syvyyskuva saadaan tekemällä useampia mittauksia eri suuntiin. Mittausnopeuden optimoimiseksi käytännön toteutus on usein yksi tai useampi nyökkäyskulmaltaan säädeltävässä tasossa pyörivä laseroptiikka. Pääasiallisena erotuksena rakenteelliseen valoon 3D-lidar siis mittaa etäisyydet pisteisiin yksi kerrallaan ja perustuu kolmiomittauksen sijaan kulkuajan suoraan tai epäsuoraan mittaamiseen.

Lidar-tekniikan hyviin puoliin kuuluu pitkä kantomatka (kymmeniä tai satojakin metrejä) ja käytännössä täydellinen riippumattomuus valaistusolosuhteista. Se on myös erittäin koeteltu ja luotettavaksi havaittu mittaamenetelmä niin maan päällä kuin allakin [17]. Suurimpana teknisenä huonona puolena lidar-tekniikan syvyysresoluutio ei aivan yllä kolmiomittausmenetelmien tasolle lähietäisyydellä. Kantomatka ja käyttötapaus huomioiden syvyysresoluutio on kuitenkin erinomainen. Käytännössä lidar-tekniikan käyttöönottoa tunnelien mittaustöissä laajemminkin rajoittaa vain laitteiston kohtuullisen kallis hinta. Käyttötapauksen kannalta voi kuitenkin nähdä positiivisena, että laserskanneri on käyttökelpoinen sensori muissakin mittaustöissä, sillä se tasaa sensorihankinnan kustannuksia.

### 3.2.3 Kulkuaikakamera

Kulkuaikakamera (engl. time of flight camera) on ominaisuuksiltaan monin puolin rakenteellisen valon ja 3D-lidarin väliltä ja näitä tuoreempi tekniikka. Se perustuu samaan mittauseriaatteeseen kuin 3D-lidar – laite lähettää valopulssin ja mittaa tavalla tai toisella aikaa sen kimpoamiseen takaisin. Erotuksena kulkuaikakamera lähettää valopulssin koko mitattavalle alueelle samalla hetkellä. Kulkuaikakameroiden useimmin käyttämä ajanmittausmenetelmä on vaihesiirtymä [16].

Hyvänä puolena kulkuaikakameran mittauseräisyys on suurempi kuin rakenteellisella valolla, vaikka vaihesiirtymän mittaaminen asettaakin tälle ylärajan. Useimmilla kulkuaikakameroilla etäisyyden yläraja on 5–7 m välillä [18]. Koko syvyyskuvan ottaminen yhdellä valotuksella myös johtaa nopeaan datankeruuseen, eikä sensorin tai mitattavan alueen liike saa mittaustosta vääristymään niin pahasti kuin piste kerrallaan mitattaessa. Huonona puolena kulkuaikakamera on herkkä monenlaisille mitattavan alueen materiaalien ja muotojen kuin myös valaistuksen aiheuttamille mittauseräistymille [18], [19]. Myös kulkuaikakameratekniikan syvyystarkkuudessa on parantamisen varaa [16].

### 3.2.4 Maatutka

Kiinnostavin muu kuin 3D-sensorivaihtoehto on esimerkiksi siltojen ja asfaltoitujen teiden kuntotarkoituksissa käytetty maatutka (engl. ground penetrating radar, GPR). Maatutka toimii altistamalla tutkittava pinta kontrolloiduille radioaalloille. Takaisinkimpoaman amplitudista ja kuluneesta ajasta kyetään erottelemaan pinnan alla olevia materiaalikerroksia ja jossakin määrin myös muotoja [20, s. 4–5]. Esimerkiksi metallitangot betonin sisässä sekä kostea ja kuiva kerrostuma samaa ainetta erottuvat menetelmällä melko helposti niiden dielektrisyyseron takia.

Menetelmän parhaita puolia on yhdellä mittauksella saatu absoluuttinen havaittujen kerrostumien paksuudesta. Huonoin puoli lienee suhteellisen lyhyt mittausetäisyys; kontaktia maatutka ei tarvitse, mutta järkevä käyttöetäisyys on silti vain kymmenien senttimetrin suuruusluokassa. Lisäksi maatutkan kyky havaita esimerkiksi metallia voi olla verkotuksen päälle tulevan betonin tapauksessa jopa mittaustulosta häiritsevä tekijä. Pistemäistä mittausta laajemman kartoituksen tekeminen myös vaatii useamman mittauksen tekemistä tunnetuista sijainneista. Esimerkiksi asfaltoitujen teiden kuntoa noin 30 cm syvyyteen asti pinnan alle valvotaan nykyisin vakionopeudella liikkuvista autoista tutkajärjestelmillä, joiden antennit ovat noin puolen metrin päässä tien pinnasta [20, s. 398]. GPR-tekniikan käyttöä on myös jo tutkittu betonikerroksen mittaamiseen tunnelissa lupaavin tuloksin [21], [22]. Tutkimuksen pohjalta suuri avoin kysymys on mittausten tulkitsemisen automatisoitavuus ja visualisoitavuus muulle kun asiantuntijakäyttäjälle.

### 3.2.5 Takymetri

Mainittakoon viimeisenä sensorityyppinä vielä takymetri, jonka ero 3D-lidariin on mittaustekniikan puolesta olematon. Käytännössä takymetri on laserskanneri, jonka pääominaisuus on lähetettyjen lasersäteiden eksaktin orientaation tunteminen suhteessa ympäristöön. Takymetriä ei olekaan varsinaisesti kehitetty pintojen näytteistämiseen vaan sijaintien mahdollisimman tarkkaan määrittelyyn itsensä suhteen kiintopisteiden avulla. Mikään ei kuitenkaan estä tuottamasta takymetrillä “3D-laserskannia” mittaamalla etäisyys riittävän moneen pisteeseen havainnoitavalla pinnalla [3].

Takymetrin parhaita puolia käyttötapauksen kannalta ovat erinomainen mittaustarkkuus sekä hinnasta riippumaton suhteellisen hyvä saatavuus, sillä takymetri on suhteellisen yleisesti käytetty mittalaite tunnelityömailla. Lisäksi takymetrillä on kiintopisteiden ollessa saatavilla mahdollista suorittaa 3D-sensoreille välttämätön skannien kohdistus itsenäisesti. Merkittävänä huonona puolena kattavan “skannin” luominen pinnasta takymetrillä on erittäin hidasta muihin 3D-menetelmiin verraten.

## 3.3 Kokonaisratkaisu

Maatutka on tarjoamansa absoluuttisen kerrospaksuustiedon takia erittäin kiinnostava vaihtoehto. Tekniikan mittausetäisyyden takia sensorin ainoa mahdollinen kiinnityspaikka on betonointialustan robottikäsivarsi; mieluiten suuttimen välittömässä läheisyydessä. Ongelmallista maatutkassa on sen tarve liikkua, mielellään

tasaisesti, lähellä mitattavaa pintaa. Käytännössä tämä voi vaatia tarkempaa robottikäsivarren operointia kuin on nykyisellä laitteistolla mahdollista. Lisäksi kerätty data on erittäin hankala käsitellä ilman, että sensorin liikettä mittaamisen aikana seurataan jatkuvasti. Esimerkiksi takymetrillä tai inertiamittarilla (engl. inertial measurement unit, IMU) tämä ehkä onnistuisi, mutta olisi joko työlästä, epäluotettavaa tai jossakin määrin molempia.

Maatutkaa kattavampaa ja helpommin käsiteltävää informaatiota saadaan 3D-sensoritekniikoilla, joiden mittausetäisyys on myös suurempi. Näiden suurin yhteinen hankaluus on tarve tehdä mittaukset suhteellisesti, eli ennen ruiskutusta mitattuun referenssitilaan verraten. Tällöin haasteeksi muodostuu tarve tehdä mittaukset täsmälleen samasta sijainnista ja orientaatiosta tai tietää referenssi- ja erotusmittausten tarkka sijainti- ja orientaatioero eli niiden välinen kohdistava jäykkä transformaatio. Tämän kohdistusongelman tekee merkittäväksi sensorin väistämätön liike ruiskutuksen aikana lähes kaikissa instrumentointiratkaisuissa; vahingossa sensori voi luonnollisesti liikkua ratkaisusta riippumatta. Sen ratkaisemalla 3D-sensoritekniikat ovat kuitenkin monin puolin ylivoimainen valinta.

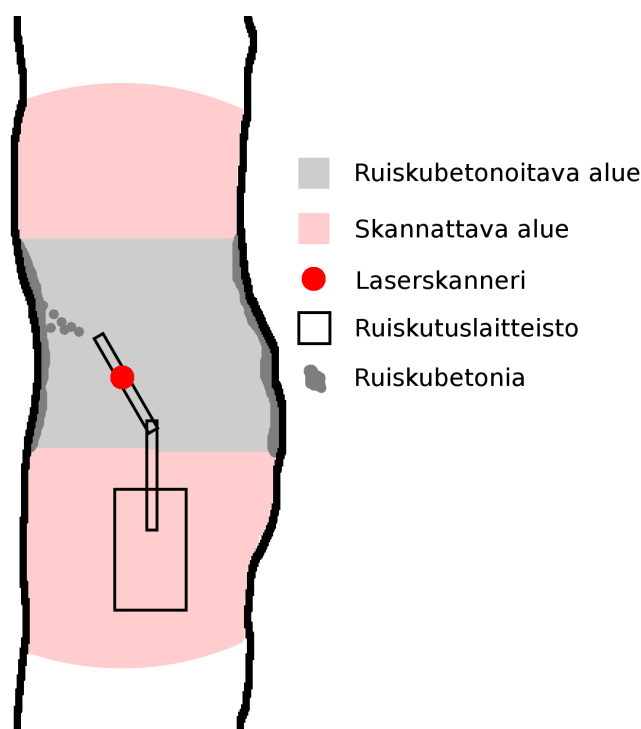
Myös 3D-sensorin liikettä voi periaatteessa seurata IMU:lla. Mittausten välinen suhteellisen pitkä aika voi kuitenkin tehdä tästä turhan epäluotettavaa IMU:n virhettä kerryttävän mittauksen takia. Toinen ratkaisu on mitata kohdistava transformaatio suoraan. Suoraviivaisin ratkaisu on tällöin takymetri, jolla on kuitenkin suoritettava monta mittausta asentotiedon selvittämiseksi. Vähemmän yksinkertainen, mutta käyttäjän kannalta vaivattomampi ratkaisu on kohdistaa skannit toisiinsa datan itsensä avulla. Tämä on mahdollista mittausalueen läheisyyteen kiinnitettyjen kiintopisteiden avulla tai vaihtoehtoisesti käyttäen hyväksi ainoastaan ruiskutetun alueen ympäröivää runsasta muuttumatonta pintaa jonkin kehittyneemmän kohdistusmenetelmän avulla. Asemoimiseen tai kiintopisteiden kiinnittämiseen kuluva aika ja tekijän asiantuntevuusvaatimus ovat merkittäviä kustannustekijöitä [2]. Käyttökelpoisimmaksi vaihtoehdoksi jää kohdistaa skannit toisiinsa automaattisesti.

Valinnassa eri 3D-sensoritekniikoiden välillä painaa eniten soveltuvuus käyttötapauksen tarkkuusvaatimuksiin, etäisyys- ja valaistusoloihin sekä automaattiseen kohdistamiseen. Karkeasti voidaan todeta, että kulkuaikakamera on tekniikkana tuorein ja erittäin kiinnostava, muttei ehkä tunneliolosuhteissa ainakaan vielä käyttökelpoinen erityisesti valaistusolosuhteiden vaihtelevuuden johdosta. Rakenteellisen valon soveltuvuus käyttötarkoitukseen taas kärsii mittausetäisyyksistä sekä tekniikan huonommasta soveltuvuudesta epätasaisten pintojen skannaamiseen, vaikka edulliset kuluttajamarkkinoiden sensorilaitteet ovatkin houkuttelevia. Laserskanneritekniikasta taas on vaikea löytää huonoja puolia hinnan lisäksi. Se tuottaa tarkkoja ja häiriöttömiä skanneja ja mahdollistaa yleisesti skannien ottamisen erittäin laajalta alueelta kerrallaan – monesti lähes täydeltä sensorikeskeiseltä pallopinnalta. Laajasta skannaussektorista on iso etu automaattista kohdistamista ajatellen. Koska takymetrikin on skannaustarkoituksessa lähinnä huono laserskanneri, on laserskanneria pidettävä parhaana sensoritekniikkana.

Kokonaisuutena käyttökelpoinen sensorityyppi ja mittaustapa on 3D-laserskanneri, jonka ottamat referenssi- ja mittausskannit kohdistetaan automaattisesti. Koska laserskannerin yhteen skanniin käyttämä aika on vähintään sekunneissa, on lasers-

kannerin pysyminen paikallaan skannauksen aikana tärkeää skannin vääristymien ehkäisemiseksi. Laadukkaat 3D-laserskannerit myös painavat yleisesti jonkin veran. Näistä syistä skanneri on parasta kiinnittää ruiskutuslaitteistoon. Kiinnitys runkoon on skannerin paikallaanpysyvyyden kannalta varma ja katveiden osalta hyväksyttävä vaihtoehto. Robottikäsiarasi taas tarjoaa minimaalisen katveen, mutta sen eksakti paikallaan pysyminen voi olla ongelma. Girmscheid ja Moser [6] ovat kuitenkin käyttäneet käsivarteen kiinnitettyä laserskanneria menestyksekkäästi tutkimuksessaan ruiskutuksen automatisoimiseksi, joten vaihtoehtoa voi pitää alustavasti parempana. Muilta osin asemat ovat käytännössä rinnasteiset. Kumpikin voi joka tapauksessa vaatia skannien kohdistamista esimerkiksi ruiskutuslaitteiston siirryttyä epähuomiossa skannien välillä. Lisäksi 3D-laserskannerien hinta, käyttöetäisyys ja katveiden määrä kummassakaan tapauksessa eivät oikeuta useamman sensoriyksikön käyttämistä. Ruiskutuslajustan robottikäsiarteen kiinnitettyyn 3D-laserskanneriin perustuva mittausjärjestelmä on hahmoteltu graafisesti kuvassa 2.

Esitetty mittausjärjestelmä on fyysiseltä osin hyvin samankaltainen automaattiruiskutuksen tutkimuksessa käytettyjen ratkaisujen kanssa (ks. [5], [6]). Näissä 3D-laserskannerin käyttötarkoitus on kuitenkin ennen kaikkea seinämän pintaprofiilin luominen automaattisesti ohjatun ”ruiskutusreitit” muodostamiseksi. Vaikka Nabulsi, Rodriguez ja Rio [5] esittävätkin skannerin käyttöä myös betonikerroksen paksuuden määrittämiseksi, he eivät määrittele tähän konkreettista ja virheitä



Kuva 2: Ehdotettu mittausjärjestelmä. Skannerin voi robottikäsiarven avulla siirtää skannaamisten ajaksi haluttuun kohtaan, esimerkiksi aivan betonoitavan alueen reunalle.

minimoivaa menetelmää. Schneider ym. [3] esittävät ruiskutuslaitteiston runkoon kiinnitettävää laserskanneria tai takymetriä nimenomaan betonikerroksen paksuuden selvittämiseksi – menetelmä kuitenkin vaatii takymetriasemoinnin tai on altis ruiskutuslaitteiston pienillekin liikkeille. Ehdotetun ratkaisun uniikki piirre on skannien automaattinen kohdistaminen, joka onnistuessaan mahdollistaa minimivirheisen paksuusmitan ilman aikaa vievää asemointia ja on robusti laitteiston pienille liikkeille.

### 3.4 Rajattu prototyyppi

Edellisessä luvussa esitetyn mittausjärjestelmän fundamentaalisimmat ongelmat ovat ennen ja jälkeen betonoinnin otettujen 3D-skannien kohdistaminen sekä mittauksen erotuksen laskeminen. Ilman näitä betonikerroksen paksuutta ei voi menetelmällä määrittää, vaikka hukkaroiskeita ja muita toimintaympäristön haasteita ei olisikaan. Näin ollen mittausjärjestelmän tietojenkäsittelyllisten ongelmien ratkaiseminen ja toimivuuden testaaminen ovat prototyypin tärkeimmät roolit.

Mittausjärjestelmän tietojenkäsittelyn ”ratkaisemisen” kannalta prototyypin on tärkeää käyttää suunniteltua sensoritekniikkaa, sillä mittaustekniikan yksityiskohdat ovat tarvittavan tiedonkäsittelyn kannalta erittäin oleellisia. Kunkin sensorityypin tuottamalla datalla on omat ominaisuutensa ja erityispiirteensä, täsmällisestä mitattavasta suuresta ja mittaustavasta riippuen. 3D-laserskanneria käyttämällä varmistetaan parhaalla mahdollisella tavalla myös kyseisen tekniikan soveltuvuus mittaustehtävään yleisestikin.

Toisaalta skannerin roiskesuojaaminen ja kiinnitys ruiskutusalueen robottikäsivarten ovat menetelmän toimivuuden testaamisen kannalta toissijaisia seikkoja. Niiden rooli kokonaisratkaisussa on taata parempi ja pidempi toimintakyky sekä minimoida katve ja tarve asiantuntijakäyttäjälle. Prototyyppiä testatessa asiantuntijakäyttäjän läsnäolo on kuitenkin välttämätöntä joka tapauksessa ja järjestelmän osoittautuessa toimivaksi skannerin suojaaminen ja kiinnittäminen ovat suoraviivaisia jatkokehitystoimia. Käytetään prototyypissä siis yksinkertaisesti siirreltävää kolmijalkaa skannerin kiinnityspisteinä. Ainoa tästä syntyvä lisäongelma tiedonkäsittelyn kannalta on skannien välinen liike, joka kohdistuksella pyritään joka tapauksessa ratkaisemaan.

Tietojenkäsittelyn ja ”lähes reaaliaikaisuuden” kannalta on lisäksi merkittävää, millaisella tietokonelaitteistolla laskentaa suoritetaan. Koska prototyypin pääasiallinen tarkoitus on testata tietojenkäsittelyn mahdollisuuksia, hyvyttä ja samalla osin suorituskykyäkin, on laskenta-alustan oltava suorituskyvyltään mieluummin liian hyvä kuin huono ”riittävään” verrattuna. Tämä ei aiheuta mittausjärjestelmän uskottavuudelle haittaa, sillä tietokoneohjelmia on mahdollista optimoida lähes poikkeuksetta huomattavastikin – ei vähiten muuntamalla algoritmeja näytönohjainpiirien suorittamaan muotoon.

Kaiken kaikkiaan toteutettavaksi mittausjärjestelmän prototyyppiratkaisuksi rajautuu kokonaisuus, jossa sensorina toimii kolmijalalle tuettu 3D-laserskanneri ja tietojenkäsittelyn roolina on kohdistaa sensorin tuottamat skannit sekä laskea niistä betonikerroksen paksuus. Tällaisen järjestelmän toteuttamalla ja sen toimintaa havainnoimalla saadaan selville konseptin yleinen toimivuus.

## 4 Tiedonkäsittely

Tiedonkäsittely ohjelmallisesti on merkittävin osa toteutettavaa mittausjärjestelmäprototyyppiä. Toteutettavan ohjelmiston päätehtävä on kohdistaa otetut skannit ja laskea niiden välinen etäisyys betonikerroksen paksuuden selvittämiseksi. Tässä luvussa käydään läpi laserskannien ohjelmalliseen käsittelyyn tarvittavia menetelmiä yleisesti sekä 3D-lidar-tekniikan erityispiirteet huomioiden.

### 4.1 Tiedonkäsittelytarve

Valittu instrumentointiratkaisu tuottaa tiedonkäsittelyjärjestelmälle kaksi 3D-skannia tunnelista; ennen ja jälkeen seinämän osan betonoinnin. Lähtökohtana voidaan pitää, että skannit on otettu karkeasti samasta sijainnista ja asennosta sekä kohtalaisen lyhyen ajan sisällä. Etäisyyttä betonoituun pintaan on betonoidun alueen yleisen koon takia todennäköisimmin kolmesta kymmeneen metriä. Betonikerroksen paksuuden selvittämiseksi skannit on kohdistettava toisiinsa, jonka jälkeen voidaan mitata pistekohtainen etäisyys pilvien välillä. Tämä vastaa pinnan siirtymää referenssi- ja mittauskannien välillä, eli betonoidun kohdan osalta betonikerroksen paksuutta. Muissakin kohdissa skanneja muutosta voi olla esimerkiksi ruiskutuksen aikana liikkuneiden ajoneuvojen tai muiden tavaroiden osalta. Valtaosan skannattua pintaa, tunnelin seinämien, pitäisi kuitenkin olla muuttumatonta ja betonoidun kohdan erottua selvästi yhtenäisenä muuttuneena alueena.

Betonikerroksen paksuudesta ollaan kiinnostuneita mahdollisimman nopeasti mittauksen jälkeen luvussa 2.3 mainitun mukaisesti. Tämän takia on olennaista, että skannien käsittelyyn ei kulu liikaa aikaa. Jotta voidaan riittävän nopeasti käsitellä skannerin tarjoamaa datamäärää, on sitä pienennettävä näytteistämällä. Lisäksi moni kolmiulotteisella datalla operoiva algoritmi suorittaa dataan ”lähimmät naapuripisteet” sekä ”naapuripisteet annetulla säteellä” -tyylisiä hakuja. Näitä voidaan nopeuttaa huomattavasti käyttämällä sopivaa avaruudellisia hakuja (engl. spatial query) tukevaa kiihdytysrakennetta.

3D-lidar on muihin 3D-sensorytyyppeihin verrattuna suhteellisen hidas tuottamaan dataa, mutta kerättyjen pisteiden laatu on hyvä. Jonkinlaista suodatusta mitatuille pisteille on tehtävä selvien mittausvirheiden poistamiseksi, mutta rajummin dataa muokkaavia menetelmiä, kuten pistenaapurustojen muotojen pehmentämistä esimerkiksi tasosovitteilla, ei liene tarpeellista suorittaa. Suotamisen sijaan sekä kohdistamisen että etäisyysmittaamisen kannalta on hyödyllistä approksimoida kullekin pisteelle sen edustaman pinnan normaali.

### 4.2 Pistepilvet

Pistepilvi on määritelmällisesti joukko pisteitä tietyssä koordinaatistossa, mikä tekee laserskannien käsittelemisestä pistepilvinä luonnollista. Kuhunkin pisteeseen voi olla liitettyinä ominaisuustietoja, kuten väri tai näytteistetyyn pinnan normaali. Merkittävä ero esimerkiksi polygoniverkkoon on, että pisteet eivät ole sidoksissa toisiinsa millään tavoin. Pistepilvillä ei näin ollen muun muassa voida kuvata pintoja yksikä-



sitteisesti. Niitä voi kuitenkin approksimoida implisiittisesti huomioimalla lähimpien pisteiden geometrian algoritmeissa tai eksplisiittisesti rakentamalla pistepilvestä polygoniverkkopinnan (ks. [23]). Tärkeä seuraus on myös pistepilvien tarve erilliselle kiihdytysrakenteelle lähimpien naapuripisteiden tehokasta etsintää varten, sillä pisteiden avaruudellista läheisyyttä ei voi välttämättä päätellä suoraan datarakenteesta itsestään.

Pistepilven määritelmä ei ota kantaa koordinaatiston tyyppiin, mutta käytännössä valtaosa aiheesta tuotetun kirjallisuuden sisällöstä on esitetty karteesisten koordinaatistojen sanelemalla tavalla. Erittäin merkittävänä seikkana myös valtaosa avoimen lähdekoodin pistepilviohjelmistoista sekä ohjelmakirjastoista, kuten Point Cloud Library (PCL) [24], Point Data Abstraction Library (PDAL) [25] sekä Cloud Compare [26] ja The 3D Toolkit (3DTK) [27] ovat suunniteltuja lähtökohtaisesti käsittelemään karteesisissa koordinaatistossa esitettyjä pistepilviä.

Mittausmekanisminsa takia 3D-lidar tuottaa skannausdatansa pallokoordinaattisena. Pistepilviä koskevan laajan tutkimuksen ja olemassa olevien ohjelmallisten apuvälineiden hyödyntämiseksi on siis syytä muuntaa skannit karteesisissa koordinaateissa esitetyiksi pistepilviksi. Muunnos on suoraviivainen ja koordinaatistojen osalta erään määritelmän mukaisesti

$$\begin{aligned}x &= r \cos \theta \cos \phi \\y &= r \cos \theta \sin \phi \\z &= r \sin \theta,\end{aligned}$$

missä  $r$  on etäisyys pallokoordinaatiston origosta,  $\theta$  korotuskulma  $xy$ -tasosta ja  $\phi$  kääntymäkulma  $z$ -akselin suhteen. Pisteisiin sidottu mahdollinen muu mittautustieto, kuten havainnon tuottanut takaisinheijastunut valomäärä, eivät luonnollisesti kaipa muunnoksia koordinaatiston vaihtuessa.

### 4.3 Kiihdytysrakenteet

Moni pistepilviä käsittelevä algoritmi tarvitsee toimiakseen tietoa kunkin pisteen lähistön muista pisteistä, mikä on luonnollinen seuraus siitä, että yksi piste 3D-avaruudessa ei ole erityisen kiinnostava. Toisaalta hyvin määritelty pistejoukko, kuten pisteen lähimmät pisteet muodostavat paikallisen kontekstin, josta on algoritmisesti pääteltävissä lisää tietoa pelkkiin pistemäisiin koordinaatteihin verraten. Esimerkiksi moneen tarkoitukseen hyödylliset pistekohtaiset normaalit on käytännössä approksimoitava lähimpiin pisteisiin tavalla toisella sovitettuun tason, pallopinnan tai jonkin eksoottisemman pinnan normaalina [28], [29].

Yksinkertaisin naapurustohakuja tehostava avaruudellinen, implisiittinen, kiihdytysrakenne on tietyissä tilanteissa pistepilven sisäinen rakenne eli pisteiden järjestys. Esimerkiksi 3D-lidarin tuottaman skannipistepilven pisteet ovat käytännössä aina tunnetussa järjestyksessä mitatun pisteen suuntaan eli skannerin optiikan orientaatioon nähden. Tällöin pistepilven datarakenteessa vierekkäiset ja jatkuvalla pinnalla sijaitsevat pisteet ovat väistämättä vierekkäiset myös 3D-avaruudessa ja naapuripisteiden etsiminen supistuu datavierekkäisten pisteiden indeksien laskennaksi  $\mathcal{O}(1)$ -ajassa [29].

Skannattavan pinnan epäjatkuvuuskohdissa tämä ei kuitenkaan päde, vaan kappaleiden reunoilla datavierekkäisten pisteiden etäisyysero on ideaalisesti etäisyys etu- ja taka-alan pintojen välillä. Pistepilven sisäinen rakenne ei kykene mahdollistamaan tehokkaita naapurustohakuja myöskään pistepilven ollessa useammasta osasta koostettu tai sen rakenteen ollessa “rikkoutunut” esimerkiksi pisteiden poissuodatuksen johdosta.

Pistepilven sisäistä rakennetta yleispätevämpi ratkaisu naapurihakujen kiihdytysrakenteeksi ovat avaruudelliset hajotelmat (engl. spatial decomposition). Niissä toisiaan lähellä olevat pisteet sijoitetaan pistepilvestä erilliseen kiihdytysrakenteeseen siten, että lähekkäisten pisteiden hakemiseen kuluva aika on alilineaarinen datajoukon kokoon nähden. Yksinkertaisimpia hajotelmia ovat 3D-avaruutta jakavat puurakenteet, kuten oktopuu (engl. octree) ja  $k$ -ulotteinen puu (engl.  $k$ -dimensional tree; KD-puu). Kumpikin osittaa avaruuden ilman aliavaruuksien liittymistä. Erottavana tekijänä oktopuu jakaa avaruuden aina kunkin akselin suhteen kerralla, eli kullakin puun solmulla (engl. node) on kahdeksan lasta. KD-puu taas on rakenteeltaan perinteisemmin binääripuu, jossa kukin solmu jakaa avaruuden kahteen osaan jonkin akselien suhteen. Eksoottisempiakin kiihdytysrakenteita on kehitetty, mutta käytännössä niille ei kolmiulotteisen datan kanssa ole tarvetta; KD-puu on kolmiulotteisen datan kanssa erittäin tehokas ja syystä käytetyin kiihdytysrakenne naapurihakussa [30]–[32].

Tarkemmin ottaen KD-puu on yleinen  $k$ :n ulottuvuuden binäärinen hakupuu, joka jakaa  $k$ -ulotteisen avaruuden ulottuvuuksien mukaan rajattuihin hypersuorakaitsiin (engl. hyperrectangle). Ulottuvuuksien lukumäärästä riippumatta puhutaan tavanomaisesti KD-puista; kaksiulotteinen esimerkkipuu on esitetty kuvassa 3 seuraavalla sivulla. Rakenteen tärkeimmät ominaisuudet ovat puun luominen pistejoukosta  $\mathcal{O}(n \log n)$  sekä naapurustohaku  $\mathcal{O}(\log n)$  ajassa [33]. Kunkin solmun jakavan ulottuvuuden ja raja-arvon valinta on KD-puun tärkeimpiä toteutusvalintoja. Vaihtoehtoja ovat muun muassa pisimmän ulottuvuuden valinta tai kaikkien ulottuvuuksien käyttäminen vuorotellen puurakenteessa syvemmälle mentäessä. Jakavana arvona voidaan käyttää esimerkiksi valitun ulottuvuuden suuntaista aliavaruuden keskikohtaa sellaisenaan tai sopeutettuna siten, että kumpaankin lapsisolmuun jää vähintään  $n_c$  pistettä. Yleisimmin käytetty jakoarvo on kuitenkin aliavaruuden pistejoukon valitun ulottuvuuden suuntainen mediaanikoordinaatti [32]. Tällä valinnalla kunkin solmun rajaamassa aliavaruudessa on terminaalitapauksia lukuun ottamatta aina puolet vanhempisolmun pisteistä ja puun syvyys pysyy minimaalisena.

## 4.4 Esikäsittelymenetelmät

Pistepilvien esikäsittelyllä tarkoitetaan pistepilville suoritettavia operaatioita ennen varsinaisesti kiinnostavaa operaatiota tai operaatioita, kuten erillisten pintojen segmentointia tai kahden pistepilven kohdistamista. Esikäsittelyn tarkoituksena on muokata pistepilveä kiinnostava operaatiota silmällä pitäen helpommin käsiteltävään muotoon pisteitä tai pistekohtaista dataa tarpeen mukaan lisäämällä tai poistamalla. Seuraavaksi käsitellään käyttötapauksessa relevantit pistepilvien esikäsittelymenetelmäkategoriat sekä olennaisimmat eksaktit menetelmät.

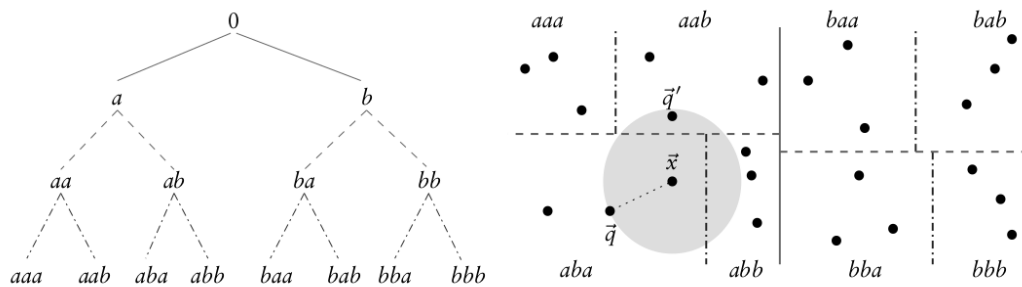
#### 4.4.1 Näytteistuksen harventaminen

Näytteistuksen harventaminen (engl. downsampling) on pistepilvilaskennan suorituskyvyn kannalta äärimmäisen kriittinen esikäsittelyvaihe [23]. Harvennetun näytteistuksen lopputuloksena pistepilvessä on vähemmän pisteitä kuin ennen näytteistystä. Pistepilveen jäävien pisteiden valinnan kriteeri on lopputuloksen kannalta tärkein määriteltävä tekijä. Karkeasti järkevät harvennuskriteerit voi jakaa datarakenteen, avaruuden tai tietosisällön mukaan tasaisesti (engl. uniform) toimiviin. Nämä vaativat järjestyksessään kasvavan määrän tietoa pistenaapurustostaan vähintään epäsuorasti.

Näytteistuksen harventaminen datarakenteen suhteen on yksinkertaisin vaihtoehto ja voidaan suoraviivaisimmillaan toteuttaa valitsemalla joka  $k$ :s piste, missä  $k$  on näytteistystaajuuden (engl. sampling rate) käänteisluku. Pallopinnan yli mitatun 3D-lidarin tuottaman skannipistepilven tapauksessa tämä vastaa skannauksen kulmaresoluution laskemista. Vastaavasti voidaan myös valita yksi satunnainen piste jokaisesta peräkkäisestä  $k$ :n pisteen joukosta, jolloin vältetään avaruudellisen laskostumisen (engl. spatial aliasing) aiheuttamia vääristymiä [25]. Tilastollisesti ottaen vielä tasajakautuneempaan näytteistykseen päästään valitsemalla pilvestä  $m$  satunnaista pistettä esimerkiksi Sanders ym. [34] esittelemillä menetelmillä.

Datarakenteen suhteen tasaisesti toimivien näytteistysmenetelmien puute 3D-lidarin tuottamien skannipistepilvien näytteistyksessä on sen säilyttämä pistetiheyden voimakas vaihtelu – kulmaresoluution ollessa vakio pistetiheys on käänteisessä suhteessa mittausetäisyyteen. Pistetiheyden vaihtelu voi kuitenkin haitata usean pistepilvalgoritmin toimintaa, erityisesti yleisesti käytettyjä kohdistusalgoritmeja [35]. Eräs ratkaisu on näytteistäminen tasaisesti avaruuden suhteen.

Vokselihilasuodin (engl. voxel grid filter) on suoraviivainen näytteistysmenetelmä, joka tasoittaa pistetiheyttä tehokkaasti [36]. Algoritmissa pistepilven peittämä avaruus kuvataan määrätyn kokoisina vokseleina ja suodatuksen läpäisee yksi piste kunkin vokselin kattamalta alueelta. Suodatuksen läpäisevä piste voi olla vokselin keskipistettä lähin pilven piste tai laskennallinen vokselin kattamien pisteiden keskipiste tai yksinkertaisimmillaan jopa vokselin itsensä keskipiste.



Kuva 3: KD-puun rakenne ja sen vuorotellen x- ja y-akselin suuntaisesti jakama avaruus. Piste  $\vec{x}$  demonstroi lähimmän pisteen hakua KD-puusta. Tässä tapauksessa lähin naapuri ei ole samasta solmusta löytyvä  $\vec{q}$  vaan avaruudellisen naapurisolmun  $\vec{q}'$ . Tästäkin huolimatta haku on ajaltaan  $\mathcal{O}(\log n)$  operaatio. [16]

Tietosisällöllinen mukaan tasainen suodatus on edellisiä eksoottisempi vaihtoehto ja lähestyy avainpisteiden (engl. keypoint) etsintää. Rusinkiewicz ja Levoy [37] esittelevät työssään pisteiden normaalien suunnan mukaan tasajakautuneen näytteistysmenetelmän, joka lienee paras tai ainakin siteeratuin esimerkki tästä näytteistysmenetelmäkategoriasta. Se säilyttää pisteitä mahdollisimman erisuuntaisilta pinnoilta ja vähentää seurauksena eniten pisteitä laajoilta tasaisilta pinnoilta. Suodatuksen lopputulos siis ei ole avaruudellisesti tasajakautunut, vaan sisältää enemmän pisteitä alueilla, joilla on vaihtelevampaa ja todennäköisesti kiinnostavampaa geometriaa. Menetelmässä siis ei käytetä avainpisteiden etsinnälle tyypillisiä jalostettuja korkeaulotteisia pistekohtaisia ominaisuustietoja (engl. feature descriptor) vaan vain kolmiulotteisia normealeja.

Huomionarvoista on, että lähtökohtaisesti näytteistäminen kadottaa pistepilvestä datarakenteeseen sisältyvän implisiittisen tiedon skannissa lähekkäisistä pisteistä. Tieto säilyy vain sopivalla datarakennepohjaisella näytteistyksellä tai vaihtoehtoisesti säilyttämällä karsitut pisteet pistepilvessä poistetuiksi merkittyinä; tällöin kaikki pistepilvelle suoritettavat jatko-operaatiot kuitenkin hidastuvat pisteiden poistuneisuustarkastelun verran. Jos siis skannin tuottaman pistepilven datarakenteesta on tarkoitus päätellä jotakin, on se ideaalisesti tehtävä ennen näytteistystä.

#### 4.4.2 Poikkeavien havaintojen suotaminen

Poikkeavalla havainnolla (engl. outlier) tarkoitetaan yleisesti muusta datasta huomattavasti poikkeavaa osaa, monesti yksittäistä arvoa. 3D-lidarilla tuotettujen pistepilvien kontekstissa poikkeavat havainnot vastaavat käytännössä mittausvirheitä ja ilmenevät pisteinä, jotka eivät selvästi sijaitse millään mitatulla pinnalla [38]. Mittausvirheitä esiintyy erityisesti pintojen epäjatkuvuuskohdissa eli kappaleiden reunoilla. Toisaalta myös jotkin materiaalit, kuten lasi tai mattamusta kangas, sekä matalat laserin tulokulmat (engl. angle of incidence) suhteessa mitattavaan pintaan, aiheuttavat vääristymiä. Tällaiset poikkeamat on tärkeää suodattaa pois pistepilvestä, sillä ne voivat haitata merkittävästi kehittyneempien pistepilvialgoritmien toimintaa [38].

Yksinkertaisimmillaan laserskannin mittausvirheet ovat tunnistettavissa suoraan skannerin raakadatasta. Kuhunkin mitattuun pisteeseen on yleisesti sidottu mittauksessa vastaanotettu heijastunut valomäärä – sen ollessa sensorilaitteistolle tavanomaiseksi määriteltyä isompi tai pienempi on kyseinen mittaus erittäin todennäköisesti virheellinen. Koska pisteiden vierekkäisyys pallopinnalla on tiedossa raakadatassa, on tältä pohjalta myös mahdollista suotaa pisteitä, joiden etäisyysero naapureidensa kanssa on liian suuri, osoittaen todennäköisen reunan etu- ja taka-alan skannauspinnan välillä. Myös huonoiksi havaittujen pisteiden viereisiä pisteitä tai esimerkiksi pisteet, joiden ympärillä ei ole yhtäkään hyvää pistettä on mahdollista suodattaa pelkän raakadatan avulla yksinkertaisesti.

Pistepilvidataa on mahdollista käsitellä myös yleisillä datapoikkeamien tunnistusalgoritmeilla, jotka Papadimitriou ym. [39] ovat luokitelleet jakauma-, syvyys-, klusterointi-, etäisyys- sekä tiheyspohjaisiin. Näistä kaksi viimeistä ovat lähtökoh- tiensa ja laskennallisen kompleksisuutensa puolesta soveltuvia pistepilvidatalle [38].

Etäisyyspohjainen poikkeamantunnistus toimii kaikessa yksinkertaisuudessaan siten, että pisteet kauempana kuin säteellä  $r$  osuudesta  $\beta$  pilven pisteitä tulkitaan poikkeamiksi. Tiheyspohjainen tunnistus taas pohjautuu  $n$  lähimmästä naapurista suoritettavaan laskentaan, joka arvioi pisteen paikallisen poikkeamakertoimen (engl. Local Outlier Factor, LOF). Suodatusta varten täytyy siis lisäksi määritellä poikkeamakertoimen hyväksyttävyyden raja-arvo  $\min_{LOF}$ . Parametrien valinta vaikuttaa kummassakin tapauksessa poikkeamien tunnistuksen hyvyyteen merkittävästi ja parametrien arvojen valinta onkin samaan aikaan tilastollissävyytteisen poikkeamantunnistuksen paras ja hankalin ominaisuus. Parametrisaation ansiosta menetelmät ovat joustavia, mutta toisaalta ne ovat erittäin helppoja ylisovittaa (engl. overfit) tiettyihin syötteisiin.

#### 4.4.3 Pintanormaalien laskenta

Pistekohtaisten pintanormaalien laskenta on lähes väistämätön pistekohtaista tietoa lisäävä eli jalostava pistepilvidatan esikäsittelyvaihe [28]. Kullekin pisteelle lasketaan normaali, jolla approksimoidaan pisteiden näytteistämän pinnan muotoa mahdollisimman hyvin. Käytännössä tämä tehdään sovittamalla kunkin pisteen naapurustoon sopiva pinta, jolta pistettä parhaiten vastaava normaali valitaan. Yksinkertaisin ja täysin toimiva pinta tarkoitukseen on taso [28]. Tasosovitteen laskeminen on tehokasta ja yksinkertaisena sovitussmallina se myöskin pitää ylisovittumisen minimissään.

Tason sovittaminen pistejoukkoon on tunnettu ongelma ja ratkaisun voi etsiä useammallakin eri tavalla. Klasing ym. [28] jaottelevat menetelmät optimoiviin sekä keskiarvoistaviin. Näistä ensimmäiset optimoivat tasosovitetta sellaisenaan ja jälkimmäiset approksimoivat tasoa naapuruston pienempien tasojen keskiarvosta. Käytännössä jälkimmäinen tarkoittaa naapuruston kolmiointia tai valmiin kolmioinnin hyväksikäyttämistä. Etenkin ilman valmista kolmiointia tasosovitteen optimointi on lähes aina parempi vaihtoehto [28].

Klasing ym. [28] jaottelevat tasosovitetta optimoivat menetelmät edelleen useampaan kategoriaan. Koska näiden kaikkien peruseriaate on sama, käsitellään tässä vain suorituskyvyltään sekä tuloksiltaan yleisesti paras pääkomponenttianalyysiin (engl. Principal Component Analysis, PCA) rinnastettava menetelmä, jonka Hoppe ym. [40] lienevät ensimmäisenä esitelleet.

PCA itsessään on tehokas, yksinkertainen ja yleispätevä menetelmä datan ulotteisuuden (engl. dimensionality) laskemiseksi [41]. Sen toiminta perustuu matriisimuotoisena esitettävän datajoukon, kuten kolmiulotteisen pistejoukon, kovarianssimatriisin ominaisarvojen selvittämiseen. Ominaisarvojen algebrallisten ominaisuuksien johdosta datajoukon varianssi on tällöin suurin ortogonaalisten ominaisvektorien kuvaamiin suuntiin. Mikäli datajoukolla havaitaan olevan vain vähän varianssia yhden tai useamman ominaisvektorin kuvaamassa suunnassa, datajoukon ulotteisuutta on mahdollista laskea rekonstruoimalla se avaruuteen, jonka kantavektoreina toimivat vain runsasvarianssiset alkuperäiset ominaisvektorit [41].

PCA-pohjainen pintanormaalien laskeminen perustuu siihen havaintoon, että kolmiulotteisen pistejoukon kovarianssimatriisin kaksi suurinta ominaisarvoa vastaavat niitä akseleita joiden suuntaisesti pistejoukolla on eniten varianssia [42]. Näitä

kahta ominaisarvoa vastaavien ominaisvektorien virittäessä tason jää tason normaaliaksi tällöin näiden molempien kanssa määritelmällisesti ortogonaalinen pienintä ominaisarvoa vastaava ominaisvektori. Tason normaalia vastaavan ominaisarvon suuruudesta voidaan myös haluttaessa päätellä tasoapproksimaation hyvyys; pienempi ominaisarvo tarkoittaa pienempää pisteiden varianssi normaalin suunnassa eli korkeampaa tasomaisuutta [41].

Normaalin naapurustopohjaista approksimoimista haastavampi ongelma on naapurustokoon valinta, sillä se vaikuttaa lopputulokseen merkittävästi ja paras mahdollinen valinta riippuu voimakkaasti käyttötapauksesta [43]. Mitä isommalta alueelta tai useampia pisteitä valitaan, sitä vähemmän normaali kuvaa pintaa pisteen kohdalla ja enemmän naapuruston keskiarvoa. Toisaalta hyvin pieneltä säteeltä valitsemalla ei pahimmassa tapauksessa löydetä edes tasosovitteelle välttämätöntä kolmea pistettä ja tulos voi painottua liikaakin lokaalin geometrian tai mittauskohinan mukaan. Hyvä lähtökohta on kuitenkin suurin mahdollinen arvo, joka mahdollista käyttötapauksen kannalta merkittävien geometrinen yksityiskohtien erottumisen [43].

## 4.5 Kohdistusmenetelmät

Kohdistamisella (engl. registration) tarkoitetaan pistepilvien kontekstissa kahden tai useamman samasta kohteesta otetun pistepilven välisen transformaation etsimistä siten, että sen läpi muunnettuina pistepilvet muodostavat yhtenäisen esityksen kohteesta (ks. kuva 4 seuraavalla sivulla) [44]. Tällöin kohdistetut pistepilvet voi haluttaessa esimerkiksi yhdistää, jolloin saadaan yksi eheä, tiheämpi ja mahdollisesti uusia pintoja sisältävä, esitys pistepilven näytteistämästä kohteesta. Toinen ja tämän työn puitteissa kiinnostavampi mahdollisuus on suorittaa pistepilvien edustamien pintojen vertailua.

Kirjallisuudessa on käsitelty lukuisia tapoja kohdistaa pistepilviä; Tam ym. [44] ovat kartoittaneet aihetta kattavasti ja huomioivat sekä jäykät (engl. rigid) että joustavat (engl. non-rigid) kohdistusmenetelmät. Tämän työn kannalta relevantimpia, kalliopinnoille hyvin soveltuvia, jäykkiä menetelmiä ovat esitelleet kattavammin muun muassa Salvi ym. [46], jotka jatkoerottelevat menetelmät karkea- ja hienokohdistaviin. Karkeakohdistuksen tarkoituksena on löytää oikeansuuntainen kohdistava transformatio pistepilvien välille mahdollisimman vähällä lähtötiedolla. Useimmiten ne pohjautuvat pisteiden korkeaulotteisten ominaisuustietojen laskemiseen ja näiden avulla tunnistettavien mahdollisimman uniikkien avainpisteiden käyttämiseen kohdistamisessa. Hienokohdistusmenetelmät taas pyrkivät minimoimaan pilvien välisen erotuksen kauttaaltaan. Ne ovat lähes poikkeuksetta luonteeltaan iteratiivisia optimointiprosesseja, joten ilman hyvää kohdistavan transformaation alkuarvoa ne voivat helposti konvergoitua paikallisiin minimeihin (engl. local minima) [46]. Transformaation alkuarvona käytetään yleisesti aiemmin suoritettua karkeakohdistuksen tulosta, odometriapohjaista tietoa tai identiteettitransformaatiota.

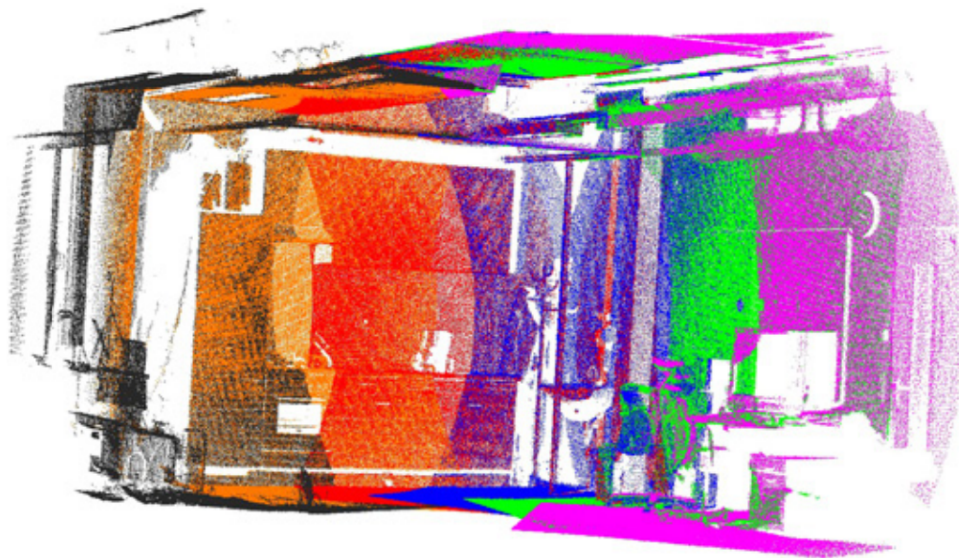
Tässä työssä ratkaistavan kohdistusongelman tärkein piirre on mittauspistepilven lähes täydellinen limittyminen referenssipilven kanssa. Poikkeamia ovat lähinnä ruiskubetonoitu alue sekä skanneihin tallentuneet, mahdollisesti liikkuneet, ajoneuvot. Lisäksi luvussa 3.3 esitetyn mukaisessa järjestelyssä pistepilven tuottava sensori on

kiinnitetty työkoneeseen siten, että lähtökohtaisesti skanneri on suhteellisen samassa sijainnissa ja orientaatiossa referenssi- ja mittausskanneja otettaessa. Karkeakohdistus ei siis ole käyttötapauksen kannalta välttämätöntä. Sen sijaan betonikerrospaksuuden senttimetrien suuruusluokan takia hienokohdistamisen on onnistuttava hyvin luotettavien tulosten saamiseksi.

Käyttötapauksen kohdistusongelman merkittävää osittaista, kohdistamisen kannalta epäkiinnostavaa, muutosta kohdistettavien pistepilvien kuvaamien pintojen välillä ei vaikuta käsiteltäneen kirjallisuudessa. Tällaisesta lähtökohdasta syntyvä potentiaalinen ongelma on kuvassa 5 seuraavalla sivulla esitetty laskennallisen ja aidon parhaan kohdistumisen mahdollinen eroavaisuus. Pistepilvien päällekkäisen muuttumattoman osan ollessa riittävästi muuttunutta aluetta suurempi on kuitenkin odotettavissa, että hyvin valittu ja sopivasti viritetty hienokohdistusmenetelmä toimii hyvin muuttumattoman alueen painaessa enemmän kohdistumisen optimoinnissa. Lisäksi menetelmästä riippuen voi olla mahdollista rajata muuttunutta kohtaa edustavat pisteet kokonaan kohdistusalgoritmin tarkastelun ulkopuolelle.

Yleisistä yhteneväisyyksistään huolimatta myös hienokohdistusmenetelmät on luonnollisesti mahdollista kategorisoida ja Salvi ym. [46] jakaakin menetelmät iteratiivisen lähimmän pisteen menetelmiin (engl. Iterative Closest Point, ICP), Chenin menetelmään [47] sekä suunnatun etäisyyskentän ja geneettisten algoritmien menetelmiin. Näiden lisäksi ainakin normaalijakaumamuunnos (engl. Normal Distribution Transform, NDT) on tunnettu, kolmiulotteisena versiona pitkälti Magnussonin väitöstyössään [16] kehittämä hienokohdistuksessa käytetty menetelmä.

Tämän työn käyttötapauksessa geneettisten algoritmien konvergoitumisaika sekä suunnatun etäisyyskentän tuloksen muoto ovat epäkäytännöllisiä. NDT:n tekee kiinnostavaksi sen mahdollisesti parempi toimivuus tunneliympäristöstä monesti



Kuva 4: Kohdistettuja pistepilviä. Kukin väri edustaa yhtä pilveä. [45]

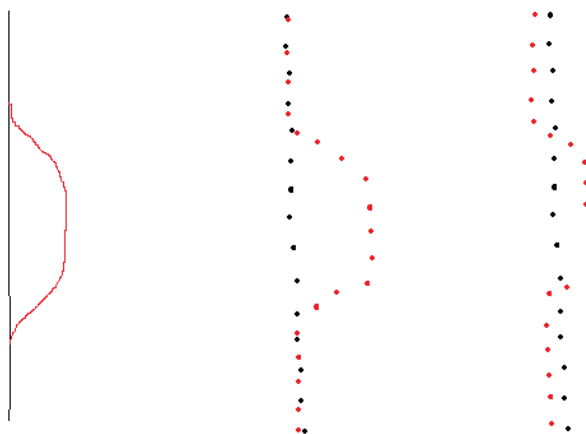
syntyvillä käytävämäisillä ja piirreköyhillä pistepilvillä [16], [48]. Se ei kuitenkaan ole yhtä koeteltu kuin ICP, eikä kohdistuksen hyvydessä tai suorituskvyvyssä ole välttämättä eroa, kun kohdistettavat pistepilvet ovat valtaosin limittäiset ja transformaation alkuarvo on hyvä [49]. ICP ja muussa kirjallisuudessa (ks. [35], [37], [44]) pääosin ICP:n variaationa pidetty Chenin menetelmä taas ovat varma ja selkeä vaihtoehto työn käyttötapauksen hienokohdistukseen.

## 4.6 Iteratiivinen lähimmän pisteen menetelmä

ICP on perinteinen, intuitiivinen ja toimivaksi todettu hienokohdistusmenetelmä. Alun perin ICP:ksi nimetyn algoritmin formalisoivat Besl ja McKay työssään yleisten kolmiulotteisten kappaleiden kohdistamiseksi [50]. Chen ja Medioni esittelivät samanaikaisesti erillään kehitetyn rungoltaan yhtenevän algoritminsa syvyyskuvien kohdistamiseksi [47]. Näiden alkuperäistöiden jälkeen rungoltaan vastaavia kohdistusmenetelmiä on kehitetty lukuisia ja kirjallisuudessa näistä puhutaan pitkälti ICP-menetelminä (ks. [35], [37]). Näkökulmasta riippuen ICP:llä siis viitataan sekä menetelmäperheeseen että alkuperäiseen “nimialgoritmiin”. Viitataan tästä eteenpäin ICP:llä menetelmäperheeseen, ellei asiayhteydessä toisin osoiteta. Käsitellään seuraavaksi ICP-menetelmiä yhdistäviä ja erottavia piirteitä sekä esitetään eri osien toiminta tarkemmin.

### 4.6.1 Algoritmin yleiskuvaus ja askeleet

ICP:n kantava ajatus on etsiä iteratiivisesti referenssi- ja mittauspistepilvien välisiä vastaavuuksia ja jäykkä transformatio, joka minimoi näiden vastaavuuksien välisen virhemetriikan [37], [51]. Selventävä pseudokoodilistaus on esitetty algoritmissa 1 sivulla 23. Alun perin ICP:ksi nimetyssä algoritmissa vastaavuuksiksi valittiin kap-



Kuva 5: Vasemmalla todellinen pinta ennen (musta) ja jälkeen (punainen) betonoinnin. Keskellä ideaalinen kohdistuminen. Oikealla laskennallisesti mahdollisesti parempi kohdistuminen



paleiden, esimerkiksi pistepilvien, lähimmät pisteet ja virhemetriikkana käytettiin näiden välisten euklidisten etäisyyksien neliöiden summaa [50]. Chenin menetelmän fundamentaalinen eroavaisuus on pistepilvien välisten etäisyyksien määrittelemineen niiden mallintamien pintojen normaalien suuntaisesti [47].

Vaikka ICP:ksi nimetty algoritmi ja Chenin menetelmä eivät korkealla tasolla eroakaan paljoa, ovat eroavaisuudet toiminallisesti sekä yksityiskohtien osalta merkittävät. Lukuisat ICP-menetelmän parannukset ovat myös muunnelleet monia algoritmin vaiheita ja jopa lisänneet vaiheita siihen. Rusinkiewicz ja Levoy [37] ovat erotelleet ICP-menetelmiä erottavat tekijät kattavasti

- käsiteltävien pisteiden valintatavaksi (engl. point selection)
- vastaavuuksien etsimismenetelmäksi (engl. correspondence estimation)
- vastaavuuksien painotusmenetelmäksi (engl. correspondence weighting)
- huonojen vastaavuuksien karsintakeinoiksi (engl. correspondence rejection)
- vastaavuuksien välisen virhemetriikan muodostustavaksi
- sekä virhemetriikan minimointimenetelmäksi,

joista ensimmäisellä viitataan niiden pisteiden valintaan, joita kullakin tai kaikilla ICP-iteraatioilla käytetään. Hyvin harva ICP-variaatio kuitenkaan käyttää iteraatiokohtaista pisteiden valintaa, joten ei käsitellä vaihtoehtoa tarkemmin. Kaikille iteraatioille yhteinen pisteiden valinta taas on määritelmällisesti esikäsittelyä ja ICP-menetelmää yleisempi ilmiö, joka on käsitelty tämän työn luvussa 4.4 tarpeellisilta osin. ICP-menetelmiä erottavia tekijöitä algoritmiin 1 seuraavalla sivulla verratessa havaitaan, että ICP:n runko on hyvin abstrakti. Käytännössä kaikki sen askeleet ovat varioitavissa ja lopputulos riippuu täysin näiden vaiheiden toteutuksesta ja parametrisoinnista.

#### 4.6.2 Vastaavuuksien etsintä

Vastaavuuksien etsintä on ICP-algoritmin iteraation ensimmäinen tosiasiallinen askel (ks. algoritmi 1 seuraavalla sivulla). Sen rooli on löytää kohdistettavien pistepilvien väliset vastaavuudet, joista rakentuvaa virhemetriikkaa minimoimalla kohdistus lopulta tehdään. Laser-skannipistepilvien ja kovergenssialueen (engl. convergence basin) kannalta erittäin merkittävänä seikkana vastaavuuksien ei tarvitse olla ”oikeita”; ICP:n iteratiivisuuden johdosta niiden riittää konvergoida kohdistusta oikeaan suuntaan. Vastaavuuksien etsinnässä tehtävät valinnat ovat erittäin tärkeitä, sillä muut algoritmin vaiheet nojaavat täysin löydettyihin vastaavuuksiin.

Besl ja McKay [50] esittelevät ICP:ksi nimetyssä alkuperäisalgoritmissaan käytettäväksi euklidisesti lähimmän pisteen valintaa, mikä lienee yksinkertaisin kohdistettavien pistepilvien välisten pistevastaavuuksien etsintämenetelmä. Chen ja Medioni [47] taas esittävät mittauspilven pisteestä sen normaalin suuntaista etäisyyttä referenssipilven tavaksi valita lähin piste referenssipilvestä. Menetelmä tunnetaan yleisesti normaaliampumana (engl. normal shooting; ks. [37], [42]), sillä intuitiivisesti

ajatellen vastaavuudeksi valitaan siinä piste, joka on samanaikaisesti mahdollisimman lähellä sekä mittauspistettä itseään että siitä “ammuttua” normaalia. Edellä mainittujen lisäksi vastaavuuksia voi etsiä ainakin erilaisilla pistepilvien paikallisten avaruuksien välisillä projektioilla tai mahdollisesti käyttäen muita pisteiden ominaisuuksia, kuten värien samankaltaisuutta tai normaalien samansuuntaisuutta, 3D-koordinaatteja rikastavana tai rajaavana tietona [37].

Tämän työn käyttötapauksessa värien käyttäminen pistevastaavuuksien etsinnässä on melko hyödytöntä tunnelin seinämän suhteellisen värityksyyden vuoksi. On myös epätodennäköistä, että mittauksen valomäärän käyttäminen olisi hyödyllistä muun muassa tunnelin pintojen vaihtelevan kosteuden ja näin ollen heijastavuuden takia. Rusinkiewicz ja Levoy [37] esittävät työssään mittauksia, joiden perusteella normaalien samansuuntaisuuden huomioiminen jo vastaavuuksia etsiessä, kuten Pulli [52] ehdottaa, saattaisi jossain tilanteessa olla hyödyllistä. Samaisten mittauksen perusteella kuitenkin vaikuttaisi siltä, että hyöty on vaihteleva, eikä lainkaan välttämättä laskenta-aikansa arvoista.

Projektiiviset menetelmät ovat vastaavuuksien etsinnässä tehokkaita, sillä ne eivät tarvitse erillistä kiihdytysrakennetta vaan eheän ja suodattamattoman syvyyskuvamaisen pistepilven. Esimerkiksi Blais ja Levine [53] esittävät yksinkertaisen käänteisen kameraprojektion käyttämistä; saman kamerakoordinaatin saavat mittaus- ja referenssipilven pisteet katsotaan vastavuuksiksi. Käyttötapauksen kannalta projektiivisten menetelmien pääongelma on tarve eheille syvyyskuvamaisille pistepilville, sillä suodattamattomuudella olisi haitallisia vaikutuksia muihin tietojenkäsittelyn vaiheisiin. Lisäksi projektion toimivuus edellyttäne turhan täydellistä alkuarvoa pilvet kohdistavalle transformaatiolle. Kaiken kaikkiaan projektiiviset menetelmät vaikuttavat käyttötapauksen kannalta lähinnä huonoilta normaaliammunnan approksimaatioilta.

```

Input: Referenssipistepilvi  $Q$ ,
        Mittauspistepilvi  $P$ ,
        Kohdistavan transformaation alkuarvo  $T_0$ ,
Output: Kohdistava transformatio  $T$ ,
 $T \leftarrow T_0$ ;
while not converged do
     $C \leftarrow \text{emptySetOfCorrespondences}()$ ;
     $W \leftarrow \text{emptySetOfCorrespondenceWeights}()$ ;
    foreach  $p_i \leftarrow P$  do
         $c_i \leftarrow \text{estimatePointCorrespondence}(T * p_i, Q)$ ;
         $w_i \leftarrow \text{weighOrRejectCorrespondence}(c_i, p_i)$ ;
    end
     $T \leftarrow \underset{T}{\operatorname{argmin}} \sum_i w_i * \text{errorMetricPerCorrespondence}(T * p_i, c_i)$ ;
end

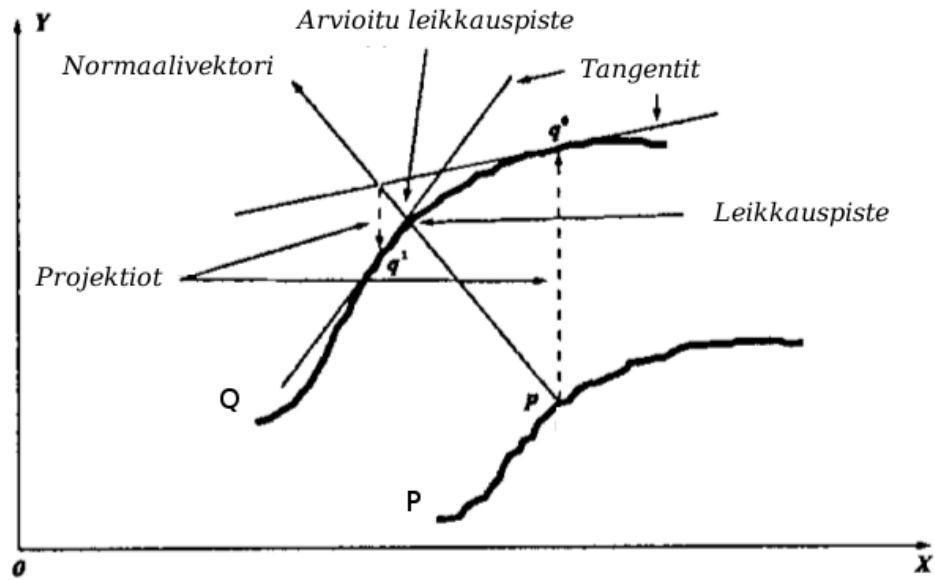
```

Algoritmi 1: ICP-algoritmin muoto ja askeleet. Selkeyden vuoksi virhemetriikka esitetty vastaavuuksittain laskettuna summana.

Tuoreempia ja eksoottisempia vastaavuudenetsintämenetelmiä edustaa esimerkiksi paikallisen kaarevuuden muutoksen käyttäminen hyväksi siten, että etsitään vastaavuuksia ensin sieltä, missä kyseinen arvo on suurin [42]. Toisaalta myös yksinkertainen lähimpien pisteiden etsiminen jonkin muun kuin euklidisen etäisyyden suhteen vaikuttaa lopputulokseen merkittävästi [35]. Näiden käytännön toimivuudesta ei kuitenkaan ole erityisen paljoa luotettavaa tietoa ja vaikka Bae ja Lichti [42] esittävätkin kaarevuusmenetelmälleen hyviä tuloksia, ovat ainoat ilmoitetut algoritmien suoritusajat minuuteissa tai kymmenissä aivan liian pitkiä tämän työn käyttötapaukseen.

Käyttötapauksen kohdistusongelmaa ajatellen parhailta vastaavuuksien etsimismenetelmiltä vaikuttavat klassiset lähimmän pisteen valinta sekä normaaliammunta. Näistä ensimmäinen on itsensä selittävä ja toimii täydellisesti yhteen kiihdytysrakenteille tyypillisesti erittäin tehokkaan lähimmän naapurin haun kanssa [32]. Normaaliammuntakin hyötyy kiihdytysrakenteesta, mutta on prosessina iteratiivinen, joten suorituskyky ei ole yhtä hyvä kuin lähimmän pisteen etsinnällä. Tarkemmin ottaen normaaliammunnassa selvitetään aina sopivan projektion, kuten sensorin havainnointisuunnan, suuntainen lähin piste  $\mathbf{q}_k$  referenssipilvessä. Tämän pisteen normaalin muodostaman tason ja mittauspilven pisteen normaalin leikkauspisteen avulla voidaan selvittää sopivampi piste  $\mathbf{q}_{k+1}$  referenssipilvestä (ks. kuva 6).

Chenin menetelmän esittelemän normaaliammunnan on moneen otteeseen todettu konvergoituvan nopeammin ja parempaan lopputulokseen lähimmän pisteen valintaan verraten etenkin kohdistavan transformaation ollessa pieni [37], [42], [46].



Kuva 6: Chenin menetelmän vastaavuuksien etsintämenetelmän eli normaaliammunnan periaate. Pisteiden  $\mathbf{q}_0$  tangentin ja  $\mathbf{p}$  normaalin avulla saadaan selvitettyä parempi leikkauspisteen approksimaatio  $\mathbf{q}_1$ . Mukaillen [47]:ta

Huomattavasti näiden tulosten yhteydessä on yleisesti käytetty myös Chenin menetelmän toista merkittävää eroavaisuutta eli älykkäämpää virhemetriikkaa. Jossain määrin epäselväksi jää, kuinka suuri osa tuloksista on syntynyt vastaavuushaun tai toisaalta virhemetriikan ja sen minimoinnin edesauttamana. Toisaalta kirjallisuudessa on viitattu usein nimenomaan Chenin menetelmän esittelemän pisteestä–tasoon-virhemetriikan käyttämiseen silloinkin, kun vastaavuuksien hakemiseksi on etsitty vastaavuuksia lähimmän pisteen menetelmällä [51] tai on nimenomaisesti kehitetty uutta menetelmää vastaavuuksien etsimiseen [42]. Kokonaisuutena voitaneen päätellä, että kumpikin menetelmä on yleisesti käyttökelpoinen.

Mainittakoon vastaavuudenetsintämenetelmistä vielä normaaliampunnan ja raa'an lähimmän pisteen etsimisen välimuoto, jollaisen toteuttaa ainakin PCL-ohjelmakirjaston `CorrespondenceEstimationNormalShooting`-algoritmi. Siinä haetaan referenssipilveltä yhdellä kertaa  $k$  naapuria, esimerkiksi kymmenen, ja valitaan näistä normaaliampuntamaisen etäisyysmitan mukaan lähin. PCL:n toteutuksessa käytetty etäisyysmitta on

$$d = \mathbf{c} \cdot \mathbf{c} = \|\mathbf{c}\|, \quad (1)$$

missä

$$\mathbf{c} = \mathbf{n} \times \mathbf{e}, \quad (2)$$

jossa vuorostaan  $\mathbf{n}$  on mittauspisteen normaali ja  $\mathbf{e}$  mittauspisteen ja sen vastaavuskandidaatin erotus. Tällöin saadaan etäisyysmitaksi ristitulon määritelmästä supistaen

$$d = \|\mathbf{e}\| \sin \angle(\mathbf{n}, \mathbf{e}), \quad (3)$$

josta nähdään, että määritelmä todella suosii voimakkaasti normaalin suunnalla olevia vastaavuskandidaatteja. Toisin kuin normaaliampunta, menetelmä kuitenkin löytää aina vastaavuuden, ellei muita rajoitteita aseteta.

#### 4.6.3 Vastaavuuksien painotus ja karsinta

Vastaavuuksien painotuksen ja karsinnan rooli ICP-algoritmissa on kasvattaa sen robustiutta antamalla vastaavuuksille painoarvoa niiden hyvyyden mukaan; konvergoitumisnopeuteen sen vaikutus on neutraali tai jopa negatiivinen [37], [54]. Etenkin kohdistettavien pistepilvien sisältäessä laajoja toisiinsa sisältymättömiä alueita on tärkeää pyrkiä painottamaan kevyemmin tai karsia täysin vastaavuuksia näiltä alueilta kohdistusprosessin vääristymien ehkäisemiseksi [37], [51]. Lisäksi kohdistavan transformaation alkuarvon huonous voi tuottaa etenkin ensimmäisillä iteraatioilla niin kaukana oikeista olevia vastaavuuksia, että ICP ei kykene konvergoitumaan oikeaan lopputilaan lainkaan ilman vastaavuuksien painotusta. Toisaalta vastaavuuksien karsiminen vähentää virhemetriikan minimointivaiheen datamäärää, mikä voi myös aiheuttaa ongelmia konvergoitumisen hyvyydessä, nopeudessa tai jopa siinä konvergoituuko algoritmi ylipäätään [54].

Huomattavasti karsintakriteerit ja painotustavat perustuvat poikkeuksetta jonkin numeerisen vertailuarvon laskentaan, ja matemaattisessa mielessä karsiminen vastaa binääristä painotusta. Periaatteessa siis mitä tahansa karsintakriteeriä voi

vertailulukunsa pohjalta käyttää painotuksena ja kaikkia painotustapoja jonkin raja-arvon kanssa karsintakriteerinä. Painotus ja karsinta eroavat toisistaan enemmän käytännöllisellä tasolla; karsintaa voi suorittaa jo osana vastaavuuksien etsintää, ja se vähentää jatkokäsitteltävän datan määrää. Kummastakin voi olla suorituskyyvyllistä hyötyä.

Yksinkertaisia karsintamenetelmiä ovat muun muassa vastaavuuksien välisen maksimivälimatkan rajoittaminen johonkin käsin asetettuun arvoon tai esimerkiksi pienemmäksi kuin jokin välimatkojen keskihajonnan monikerta [35]. Näistä etenkin ensimmäinen on erittäin yleisesti käytetty tapa ehkäistä pistepilvien osittaisen limitäisyydet (engl. partial overlap) aiheuttamia kohdistusongelmia [37], [51], [54]. Se on myös mahdollista suorittaa erittäin tehokkaasti osana useiden eri kiihdytysrakenneiden naapurustohakuja [32]. Keskihajonnan monikerran käyttäminen etäisyysrajana kiinteän etäisyyden sijaan kuulostaa erittäin elegantilta vaihtoehdolta, mutta käytännössä vaikutus on pieni [37]. Tilastollisten tunnuslukujen laskeminen vaatii myös vähintään yhden ylimääräisen vastaavuusjoukon läpikäynnin, eli mahdollisesti merkittävän määrän aikaa.

Myös ominaisuustietojen käyttö karsintakriteerinä on luonnollisesti mahdollista. Vastaavuuksien normaalien samansuuntaisuutta tai värien yhteneväisyyttä samoin kuin arvioitua mittausvirhettä voi käyttää tässä ICP:n vaiheessa siinä missä jo vastaavuuksia etsiessäkin [37]. Huomattavaa on, että tällöin vastaavuusjoukon koko kärsii, mutta vastaavuushaut on mahdollista tehdä epäiteraatiivisesti ja näin tehokkaammin. Näennäisestä suorituskyykyhyödyistä huolimatta kompromissi saattaa olla huono vastaavuusjoukon mahdollisesti merkittävän pienenemän johdosta [55]. Lisäksi tässä työssä käyttökelpoisimman ominaisuustiedon eli normaalien samansuuntaisuuden osalta käytännön vaikutus ICP:n toimivuuteen on todettu vähäiseksi [37].

Eräs mahdollisesti merkittävän robustiushyödyn tuottava keino on myös vähemmän painon antaminen skannissa yhtenäisen pinnan ääriiviivaa kuvaavan pisteen sisältäville vastaavuuksille [37], [54], [56]. Samoin kuin vastaavuuksien maksimivälimatkalla, tällä pyritään ensisijaisesti eliminoidaan kohdistettavien pistepilvien osittaisen limittymisen aiheuttamat systemaattiset virheet kohdistumisessa. Tällaisen systemaattisen virheen syntymekanismi on esitetty kuvassa 7 seuraavalla sivulla. Turk ja Levoy [56] esittivät alun perin näiden pisteiden tunnistamista pistepilven kolmioinnissa syntyneen kolmioverkon reunaa hyväksikäyttäen (ks. punaiset ja siniset katkoviivat kuvassa 7), mikä on tämän työn kannalta turhan raskasta. Ne ovat kuitenkin mahdollisia havaita myös implisiittisesti saman referenssipilven pisteen käytöstä useamman mittauspilven pisteen vastaavuutena [54]. Tällöin vastaavuuksia voidaan säilyttää enemmän pitämällä paras näistä vastaavuuksista (ks. siniset katkoviivat kuvassa 7).

Tämän työn kannalta tärkein karsinnassa huomioitava asia lienee pistepilvien suhteellisen täydellinen limittyminen luvussa 3.3 esitetysti. Ruiskubetonoidun alueen lisäksi pistepilvien välisiä mahdollisia epälimittymiä voivat aiheuttaa lähinnä ruiskutuksen aikana liikkuneet ajoneuvot, jotka erottuvat erittäin selkeästi. Todennäköisesti yksinkertainen käsin asetettu vastaavuuksien maksimietäisyys on riittävä ja suorituskyyvyltään paras ratkaisu. Käytettävän raja-arvon valinta on tunnetusti painotusvalinta konvergoitumisnopeuden ja lopputuloksen tarkkuuden välillä [51].

#### 4.6.4 Virhemetriikan minimointi ja konvergenssi

Kunkin ICP-iteraation lopuksi suoritetaan määritellyn virhemetriikan minimointi. Edeltävien iteraation askeleiden jälkeen vaiheen rooli on etsiä vastaavuuskohtaisesti painotetun virhemetriikan minimoiva uusi kohdistava transformaatio. Kun tämä toistetaan kullakin iteraatiolla aina uudelleen valituilla ja painotetuilla vastaavuuksilla, löydetty kohdistava transformaatio konvergoituu lähemmäs virhemetriikan sanelemaa laskennallisesti parasta arvoaan. ICP-menetelmän toimivuus perustuu tämän arvon lähes poikkeuksetta riittävään läheisyyteen ”oikean” transformaation kanssa, sillä esimerkiksi kahta eri laserskannia ei ole edes mahdollista kohdistaa absoluuttisen oikein niiden tuottaman pintojen erilaisen näytteistysten takia. Lopuksi tarvitaan konvergenssiehto ICP-algoritmin iteraation lopettamiseksi.

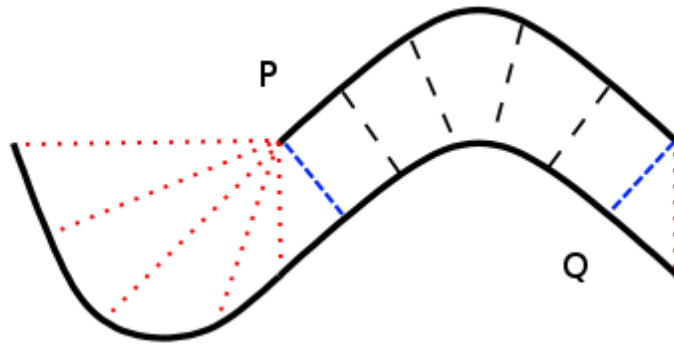
Perinteisiä ratkaisuja virhemetriikaksi ovat klassisen ICP:n [50] käyttämä vastaavuuksien pisteestä–pisteeseen-etäisyyksien neliöiden summa

$$E_{ICP} = \sum_i \|\mathbf{p}_i - \mathbf{q}_i\|^2 = \sum_i (\mathbf{p}_i - \mathbf{q}_i) \cdot (\mathbf{p}_i - \mathbf{q}_i) \quad (4)$$

sekä Chenin menetelmässä [47] käytetty neliöllisten pisteestä–tasoon-etäisyyksien summa

$$E_{Chen} = \sum_i (\mathbf{p}_i - \mathbf{q}_i) \cdot \mathbf{n}_i, \quad (5)$$

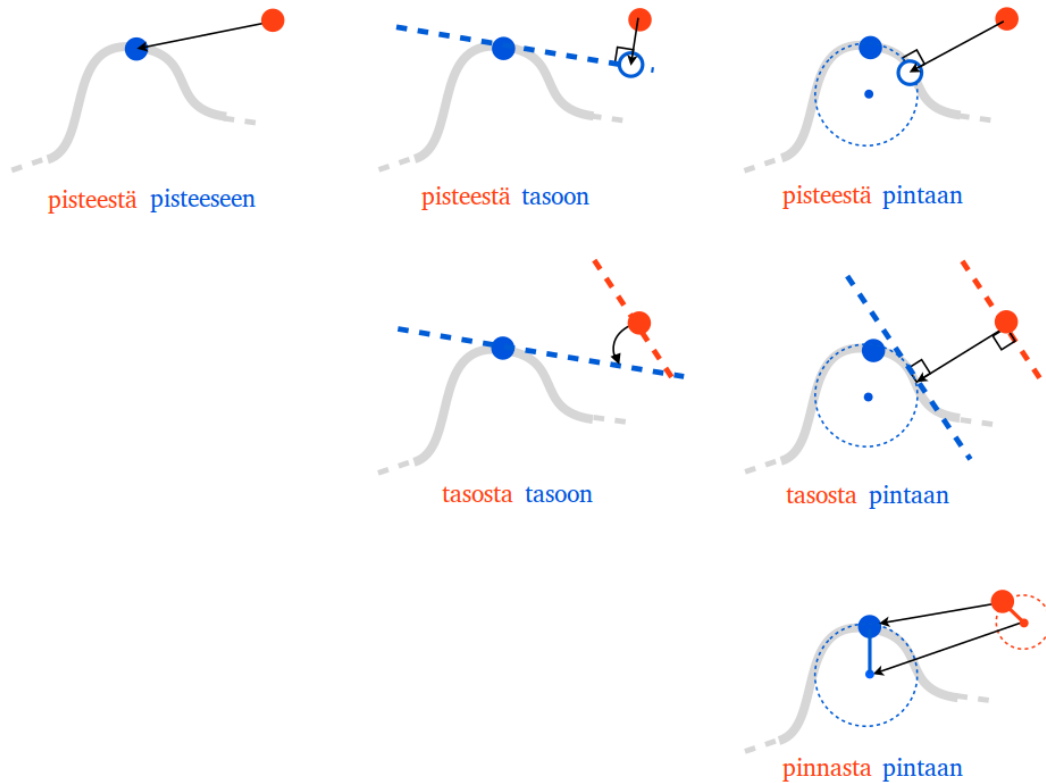
joissa  $\mathbf{p}_i$  on mittauspilven piste,  $\mathbf{q}_i$  sitä vastaava referenssipilven piste ja  $\mathbf{n}_i$  referenssipisteen normaali. Vaikka näiden ero on näennäisesti pieni, ensimmäinen tuottaa lineaarisen optimointiongelman ja jälkimmäisen taas epälineaarisen [37]. Pisteestä–pisteeseen-virhemetriikan minimoimille löytyy siis analyyttinen ratkaisu, siinä missä pisteestä–tasoon-metriikka sanelee epälineaarisen optimointimenetelmän käyttämisen



Kuva 7: Pintoina esitetyt mittauspilvi  $\mathbf{P}$  ja referenssipilvi  $\mathbf{Q}$ . Mittauspilven yhtenäisen pinnan ääripisteiden parhaat vastaavuudet merkattu sinisellä ja tiheällä, loput punaisella ja harvalla, katkoviivalla. Mustat katkoviivat esittävät muita piste-vastaavuuksia. Kaikkien vastaavuuksien virheen minimointi johtaa kohdistumiseen huomattavasti liikaa vasemmalle. Mukaillen [37]:ta

tai ongelman lineaarisen approksimoinnin. Tässä tapauksessa linearisointi merkitsee olettaa hyvää kohdistavan transformaation alkuarvosta eli pienestä rotaatiovirheestä [47].

Pisteestä-pisteeseen- ja pisteestä-tasoon-etäisyyksiä huomattavasti eksoottisempienkin etäisyysmittojen neliösummia on ehdotettu käytettäväksi virhemetriikoina [35]. Näistä merkittävin lienee Generalized-ICP -algoritmin [51] yhteydessä esitelty konseptuaalinen tasosta-tasoon-etäisyys. Tärkeimmät 3D-avaruudessa toimivat metriikat on selkeyden vuoksi esitetty kuvassa 8. Näiden lisäksi muun muassa erilaisten entropioiden käyttämistä osana etäisyysmetriikkaa, samoin kuin täysin etäisyyksien neliöiden summa -formalismin ulkopuolisia metriikoiden käyttämistä on tutkittu [35]. Näitä eksoottisempia menetelmiä ei kuitenkaan tunnu juuri käytettäneen tai vertailun kirjallisuudessa esittelynsä lisäksi, joten niiden toimivuutta tai suorituskkyä suhteessa muihin vaihtoehtoihin on vaikea arvioida. Klassisten pisteestä-pisteeseen- ja pisteestä-tasoon-etäisyyksien käyttöä on sen sijaan käsitelty runsaasti, joten keskitytään niihin.



Kuva 8: Eri vaihtoehtoja pistevastaavuuksien väliseksi etäisyysmitaksi. Punainen kuvaa mittauspilven ja sininen referenssipilven pistettä, nuoli taas suunnattua etäisyysmittaa. Mukailleen [35]:ta

Pisteestä–pisteeseen-etäisyyksien neliösumman (4) tuottama minimointiongelman voi esittää ilman vastaavuuksien painoja esimerkiksi muodossa

$$\mathbf{T} = \operatorname{argmin}_{\mathbf{T}} \sum_i \|\mathbf{T}\mathbf{p}_i - \mathbf{q}_i\|^2 = \operatorname{argmin}_{\mathbf{R}, \mathbf{t}} \sum_i \|\mathbf{R}\mathbf{p}_i - \mathbf{q}_i - \mathbf{t}\|^2, \quad (6)$$

missä  $\mathbf{T}$  kohdistava transformaatio yhden iteraation sisällä sekä  $\mathbf{R}$  ja  $\mathbf{t}$  sen ositus rotaatioksi ja translaatioksi. Kuten jo mainittua, kyseinen minimointiongelma on mahdollista ratkaista analyyttisesti. Useista mahdollisista ratkaisuksista eräs laajasti käytetty on vastaavuusjoukon ristikovarianssin pääakselihajotelmaan (engl. singular value decomposition, SVD) pohjautuva menetelmä, jonka esittelivät Arun, Huang ja Blostein [57]. Ratkaisun johto ei ole tässä yhteydessä kiinnostava, mutta perustuu rotaatio- ja translaatiokomponenttien erottamiseen toisistaan. Ratkaisu itse on

$$\begin{cases} \mathbf{R} = \mathbf{V}\mathbf{U}^T \\ \mathbf{t} = \mu_{\mathbf{Q}} - \mathbf{R}\mu_{\mathbf{P}}, \end{cases} \quad (7)$$

jossa  $\mu$ :t ovat pistepilvien keskipisteet ja  $\mathbf{U}$  sekä  $\mathbf{V}$  osa vastaavuuskovarianssimatriisiin SVD-hajotelmaa  $\mathbf{\Sigma} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$ .

Pisteestä–tasoon-etäisyyksien neliösumman (5) tuottama optimointiongelma on esitettävissä samaan tapaan kuin pisteestä–pisteeseen-variaation (6) muodossa

$$\mathbf{T} = \operatorname{argmin}_{\mathbf{R}, \mathbf{t}} \sum_i \|(\mathbf{R}\mathbf{p}_i - \mathbf{q}_i - \mathbf{t}) \cdot \mathbf{n}_i\|^2. \quad (8)$$

Kuten mainittua, tämän ratkaiseminen vaatii rotaation kulmat pieneksi olettavaa linearisointia

$$\mathbf{R} = \mathbf{R}(\alpha, \beta, \gamma) \approx \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix},$$

jossa  $\alpha$ ,  $\beta$  ja  $\gamma$  ovat osarotaatiot kunkin akselin suhteen ja eksaktin rotaatiomatriisin sini- ja kosinifunktiot on supistettu tavanomaisesti pienet kulmat olettaen. Tällöin kohdistavan transformaation parametrit  $\tau = [\alpha \ \beta \ \gamma \ \mathbf{t}^T]^T$  ovat saatavilla tehokkaasti ratkaisemalla lineaarinen järjestelmä

$$\mathbf{G}\mathbf{G}^T\tau = \mathbf{G}\mathbf{h}, \quad (9)$$

jossa  $\mathbf{G}$  ja  $\mathbf{h}$  ovat vastaavuuksista ja referenssipilven normaaleista vastaavuukskohtaisen vähennyslaskun, pistetulon ja ristitulon avulla kasatut  $6 \times N$  ja  $N \times 1$  matriisit [35]. Vaihtoehtoisesti optimointiongelman voi ratkaista jollakin numeerisella menetelmällä, kuten Levenberg-Marquardt [37]. Linearisointi on kuitenkin pisteestä–tasoon-etäisyyksien neliön yhteydessä vakiintunut ratkaisu, sillä ICP:n iteratiivisuus antaa anteeksi approksimoinnin aiheuttamaa virhettä [35]. Lineaariapproksimaation pätevyyden rajoja lähestyttäessä ICP:n kyky konvergoitua oikeaan lopputulokseen on muidenkin tekijöiden kannalta koetuksella.

Neliöllisten pisteestä–pisteeseen- ja pisteestä–tasoon-etäisyyksien minimoinnin ratkaisujen laskennallinen vaativuus on karkeasti samalla tasolla. Pisteestä–tasoon-minimointi kuitenkin vaatii enemmän liukulukuoperaatioita muun muassa  $\mathbf{G}$ :n kerrotaskuihin sekä kuuden yhtälön lineaarijärjestelmän ratkaisemiseen. Sen on silti



havaittu konvergoivan ICP-algoritmi lähes poikkeuksetta vähemmällä iteraatioilla, lyhyemmässä ajassa ja lähemmäs globaalia minimiä [37], [46]. Ylivertaisuus näkyy etenkin kohdistavan transformaation alkuarvon ollessa kohtuullisen hyvä [58]. Suurelta osin paremmuus selittyy pisteestä-tasoon-etäisyyden sallimalla mittauspilven “liukumalla” referenssipilven lokaaleilla tasoilla [37], [52]. Tämä johtaa suuremman kohdistavan transformaation parannuksen löytämiseen yhdellä iteraatiolla, jolloin konvergenssi saavutetaan vähemmällä iteraatioilla. Ominaisuudesta voi kuitenkin olla erittäin tasomaisia pistepilviä kohdistuessa myös haittaa [16], [52].

Käytetystä virhemetriikasta riippumatta ICP on iteratiivinen ja numeerinen menetelmä, joten se tarvitsee konvergenssiehdon pysähtyäkseen. Mielenkiintoisesti sen konvergoituvuutta ylipäättään ei ole formaalisti todistettu muuten kuin alkuperäisen nimimenetelmän esittelyn yhteydessä (ks. [50]) ja todistus lakkaa pätemästä esimerkiksi vastaavuuksia karsittaessa [54]. Käytännössä ICP-variaatioiden on todettu konvergoituvan hyvin, vaikka kirjallisuudessa onkin kiinnitetty suhteellisen vähän huomiota käytettyyn konvergenssiehtoon. Eräitä mahdollisia ehtoja ovat neliöllisen keskivirheen tai transformaation suuruus joko alkutilanteeseen verraten tai iteraatioiden välillä sekä iteraatioiden kokonaismäärä [59]. Molemmat alkuperäistyöt käyttävät konvergenssiehtona absoluuttista neliöllistä keskivirhettä, joka onkin verrannollinen reaali maailman kannalta relevantin kohdistustarkkuuden kanssa.

## 4.7 Pistepilvien välinen erotus

Määrittämällä kahden samasta kohteesta otetun kohdistetun pistepilven välinen pistekohtainen etäisyys määritetään samalla niiden kuvaamien pintojen välinen eroavaisuus. Työn käyttötapauksessa saadaan tällä tavoin määritettyä vastaruiskutetun betonikerroksen paksuus, kun pistepilvet vastaavat luvussa 3.3 esitetyllä tavalla otettuja skanneja tunnelista. Vaihtoehtoisia tapoja määritellä etäisyys on useita kuten jo ICP:n virhemetriikoita käsitellessä todettiin (ks. luku 4.6.4). Pistepilvien erotuksen kannalta etäisyyksien kannalta tärkein huomioon otettava seikka on pistepilven diskreetti olemus. Äärellisen kokoisena näytejoukkona jatkuvasta pinnasta se ei voi ideaalisesti edustaa kuvaamaansa pintaa.

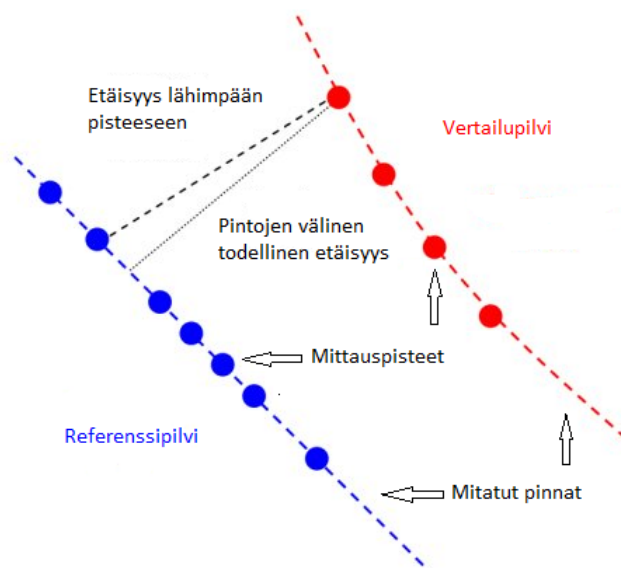
Yksinkertaisin pisteestä-pisteeseen-etäisyys kuvaa pistepilvien eroamaa toisistaan ennen kaikkea joukkona näytteitä. Sen tarkkuus pintojen etäisyyden mittana nousee näyte- eli pistetiheyden kasvaessa. Pisteestä-pisteeseen-etäisyys on kuitenkin aina pintojen välistä etäisyyttä suurempi lokaalin tasomaisuuden olettaen (ks. kuva 9 seuraavalla sivulla). Erityisen vääristynyt mitattu arvo on pintojen välisen etäisyyden ollessa lähellä nollaa, jolloin laskennallinen etäisyys kuvaa enemmän referenssipistepilven tiheyttä kuin pintojen välistä etäisyyttä. Riittävän suurella mittauksella menetelmä voi kuitenkin pitää tarkkana, vaikka se silti onkin vähimmäispaksuuden tunnistamisen kannalta epäideaalinen.

Paremmen estimaatin nimenomaan pistepilven näytteistämien pintojen väliselle etäisyydelle voi saada approksimoimalla paikallisesti referenssipistepilven kuvaamaa pintaa, erotusmittauksen pistepilven kuvaamaa pintaa tai molempia. Käytännössä referenssipinnan approksimointi lienee riittävä vaihtoehto, jonka toteutustavaksi Girardeau-Montaut ym. [61] ehdottaa naapuruston tasosovitetta, Delaunay-

kolmiointia tai splinipintasovitetta. Näistä tasosovite vastaa aiemmin esiteltyä pisteestä–tasoon-etäisyyttä (ks. luku 4.6.4), joka lienee vaihtoehtoista paras tehokkuutensa ja ylisovittuvuusturvallisuutensa vuoksi. Pisteestä–tasoon-etäisyyttä käyttäessä etenkin tasaiset pinnat saavat erittäin lähellä oikeaa olevan arvon pintojen erotukselle. Toisaalta pinnaltaan kaarevilla alueilla saadaan väistämättä todellista pintojen välimatkaa pienempi mittaustulos, sillä pisteestä–tasoon-etäisyys  $\mathbf{n} \cdot (\mathbf{p} - \mathbf{q})$  on normaalin  $\mathbf{n}$  yksikköpituisuus olettaen väistämättä sama tai pienempi kuin pisteestä–pisteeseen-etäisyys  $\|\mathbf{p} - \mathbf{q}\|$ .

Käytetystä etäisyysmitasta riippumatta laserskannaamalla saatujen pisteiden väliin voi periaatteessa jäädä millaista pintageometriaa tahansa. Tätä on mahdotonta estää ja ainoa keino ehkäistä mahdollisia kattavuusongelmia on käyttää riittävän suurta skannerin mittaustiheyttä sekä pistepilvien näytteistykseen jälkeistä pistetiheyttä. Käytännössä käyttötapauksessa mitatut pinnat ovat kuitenkin melko tasomaisia etenkin paikallisesti.

Mitattujen pintojen paikallinen tasomaisuus tekee pisteestä–tasoon-etäisyyden käyttämisestä erittäin houkuttelevaa. Pisteestä–pisteeseen-etäisyys on ylipäätään käyttökelpoinen vaihtoehto vain, jos laskennan tehokkuus saadaan pidettyä kohtuullisena korkeatiheyksilläkin pistepilvillä; riittävä pistetiheys lienee joitain millimetrejä ruiskubetonin kerrospaksuuden senttimetrien suuruusluokasta johtuen. Pisteestä–tasoon-etäisyyden käyttöä puoltaa myös se, että lähtökohtaisesti lienee parempi raportoida liian pieni kuin suuri paksuus.



Kuva 9: Pintojen ja pistepilvien etäisyyden ero. Mukailten [60]:ta

## 5 Prototyypijärjestelmä

Luvussa 3.3 esitetyn 3D-laserskanneriin pohjautuvan mittausjärjestelmän toimivuuden testaamiseksi toteutettiin luvussa 3.4 rajatun mukainen prototyypijärjestelmä. Tässä luvussa esitellään prototyypijärjestelmässä käytetty laitteisto sekä toteutettu mittausohjelmisto. Lisäksi käydään läpi prototyyppiä testatessa tehdyt havainnot ja esitetään joitain jatkokehitysmahdollisuuksia.

### 5.1 Laitteisto ja ohjelmistoalusta

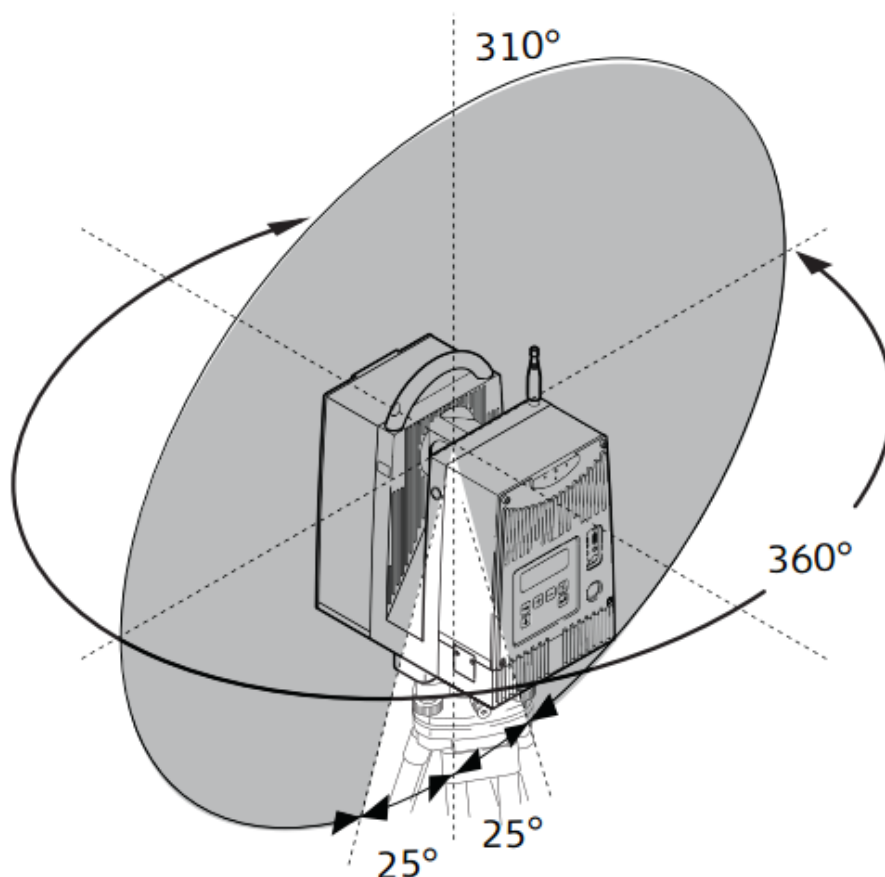
Prototyypijärjestelmän 3D-laserskannerina käytettiin Leica HDS6200 -skanneria kolmijalkatuennalla (ks. kuva 10). Se käyttää näkyvän valon aallonpituuksia 650–690 nm ja mittaa laserin kulkeman etäisyyden vaihesiirtymästä. Etäisyyden monitulkintaisuuden raja on laitteistolla 79 m ja valmistajan ilmoittama mitattavissa oleva äärietäisyys osumapinnan heijastuvuudella (engl. albedo) 90 % on 79 m ja 18 %:lla 50 m. Mitatun pisteen sijainnin luvattu tarkkuus (engl. accuracy) on 25 m päähän skannerista alle 5 mm ja 50 m asti alle 9 mm. Pinnan ja pinnanmuutosten mittaamisen kannalta kriittinen lähekkäisten pisteiden sijainnin suhteellinen mitaustarkkuus eli karkeasti ottaen pisteiden sijainnin toistotarkkuus (engl. precision) on 25 m päässä  $\delta$  2 mm ja 50 m päässä  $\delta$  4 mm. [62]



Kuva 10: Leica HDS6200 3D-laserskanneri. [62]

Leica HDS6200 mahdollistaa skannien ottamisen koko pallopinnan lisäksi myös rajoitetulta 3D-sektorilta sekä viidellä eri kulmaresoluutiolla. Skannatun sektorin rajoittaminen ei ole käyttötapauksessa mielekästä sen aiheuttaman muuttumattoman pinnan häviön johdosta. Kulmaresoluutioasetuksella sen sijaan on merkittävä vaikutus skannausaikaan, tuotetun datan määrään sekä näytteistetyn pinnan mallinnustarkkuuteen. “Valmiin” mittausjärjestelmän käytettävyyden kannalta olisikin tärkeää kokeellisesti selvittää pienin mahdollinen tarkkuudeltaan riittävä kulmaresoluutio skannausajan minimoimiseksi. Mittausmenetelmän toimivuuden testaamisen kannalta kyseessä on kuitenkin merkityksetön optimointi, joten prototyypin käyttöön valittiin ilman erityistä testaamista riittäväksi arvioitu kulmaresoluutio. Luvussa 3.3 esitetyllä noin 10 m mittausetäisyydelle valinta tuottaa noin  $25000 \text{ m}^{-2}$  pistetiheyden eli karkeasti yhden pisteen jokaiselle pinnan  $6 \text{ mm}^2$ :lle. Skannausaika täyden pallopinnan yli asetuksella on noin kolme minuuttia.

Kuten nykyisin käytännössä kaikki muutkin 3D-lidarit, Leica HDS6200 tuottaa datansa kolmiulotteisuuden mittaamalla pistekohtaisia etäisyyksiä optisesta keskuksesta ympäristöönsä sensorin suuntauksen kulkiessa pallopinnan yli. Tällöin



Kuva 11: Leica HDS6200:n näkökenttä. Runko pyörii pysty- ja optiikka vaaka-akselin ympäri. [62]

kullekin mitatulle pisteelle saadaan tieto sen etäisyydestä sensoriin yhdessä sensorin mittauksenaikaisen asennon kanssa. Asennon parametrisoivat kulmat pysty- ja vaakakselien ympäri; niin sanotut kääntymä- ja nyökkäyskulmat (engl. yaw tai heading ja pitch). Käytetyssä mallissa pysty akselin ympäri pyörii sen runko, vaakakselin taas optiikka. Tämä rakenne tarjoaa skannerille lähes täyden pallopinnan näkyvyyden. Skannerin näkökenttä ja asennon parametrisoivat kulmat on esitetty kuvassa 11 edellisellä sivulla. Etäisyyden ja optiikan asennon lisäksi skanneri tallentaa myös pistekohtaisen mittauksessa käytetyn valon määrän. Raatassa muodossaan skanneri tuottaa siis pistepilviä pallokoordinaatistossa yhdessä kuhunkin pisteeseen sidotun valomäärän kanssa.

Vaikka laserskanneri on prototyypin datalähteenä sen monella tapaa tärkein komponentti, on myös käytetyllä tietokonelaitteistolla ja ohjelmistoympäristöllä merkitystä etenkin laskennan suorituskyvyn kannalta. Prototyypin laskenta-alustana käytettiin tietokonetta, jossa on 16 Gt käyttömuistia ja Intelin neliytminen, 2,6–3,5 GHz taajuudella toimiva, i7-6700HQ -suoritin. Laskentasuorituskyvyltään laitteisto on siis kuluttajamarkkinoiden paremmasta päästä ja hintatasonsa puolesta paremmallakin ”valmiissa” järjestelmässä korvattavissa. Ohjelmistoympäristönä käytettiin Microsoftin Windows 10 -käyttöjärjestelmään asennettua Visual C++ 2015:tä.

## 5.2 Mittausohjelmisto

Toteutettu ohjelmisto vastaa rakenteeltaan ja vaiheiltaan luvussa 4 esitettyä. Sensorin tuottamat ennen ja jälkeen betonoinnin otetut laserskannit muunnetaan pistepilviksi, joista huonot pisteet suodatetaan pois. Tämän jälkeen pistepilvistä näytteistetään pistetiheydeltään matalammat pilvet, jotka kohdistetaan toistensa kanssa. Kohdistuksen jälkeen lasketaan pistepilvien välinen pistekohtainen etäisyys ja visualisoidaan tämä käyttäjälle etäisyyden mukaan värjättyä pistepilvenä, josta ruiskubetonikerros erottuu yhtenäisenä suuremman etäisyyden alueena. Kaikissa ohjelmiston osissa hyödynnetään vahvasti lukuisia pistepilvialgoritmeja ja -visualisointiapuvälineitä sisältävää PCL-ohjelmakirjastoa [24]. Kuvataan seuraavaksi tarkemmin ohjelmiston eri osat sekä algoritmit ja käytetyt parametrit.

### 5.2.1 Skannien muuntaminen pistepilviksi

Mittausohjelmiston ensimmäinen vaihe on skannien muuntaminen pistepilviksi. Käytännössä tämä tarkoittaa skannerin tuottamien skannitiedostojen lukemista ja niiden sisällön tulkitsemista ja kopioimista ohjelmaan käyttämään PCL:n määrittelemään pistepilviformaattiin. PCL:n pistepilviformaatin käyttäminen on sekä käytännöllinen että tehokas valinta, sillä runsaasti hyödynnetyt PCL:n algoritmit toimivat joka tapauksessa sen avulla.

Käytetty laserskanneri tukee vain valmistajan omaa, suljettua, zfs-tiedostomuotoa skannien säilömiseen ja siirtämiseen, joten skannien käsittelyyn tarvitaan valmistajan tarjoamaa sovelluskehityspakettia (engl. Software Development Kit, SDK) [63]. Tämän ZF-SDK:ksi nimetyn paketin toiminnallisuuksilla voi muun muassa lukea skannin pisteitä sekä optiikkakeskeisessä karteesisessa- että pallokoordinaatistossa ja

suodattaa pisteitä verrattain yksinkertaisilla menetelmillä. Konkreettisesti ohjelmiston ensimmäinen askel onkin lukea ZF-SDK:n avulla skannitiedostoista karteesisissa koordinaateissa pisteet, joiden valomäärä on valmistajan käytetylle skannerimallille asettamien rajojen sisällä. Luetuista pisteistä luodaan PCL-pistepilvi, joka siis on epärakenteellinen ja sisältää vain mittauksen valomäärän perusteella laadultaan hyvät pisteet.

### 5.2.2 Esikäsittely ja kiihdytysrakenteet

Useamman skannin silmämääräisellä visuaalisella tarkastelulla ZF-SDK:n suorittaman valomääräpohjaisen suodatuksen lopputulos oli hyvä eli virheellisesti mitattujen pisteiden määrä pistepilvissä vaikutti vähäiseltä. Ohjelmistoon ei näin ollen ajoajan minimoimiseksi toteutettu muuta esikäsittelyä poikkeavien havaintojen suodattamiseksi pistepilvistä. Lisäksi kaikki esikäsittelyvaiheet suoritetaan yksinkertaisuuden vuoksi samanlaisina sekä referenssi- että mittauspilville.

Kohdistuksen vaatimaa aikaa ja pisteiden avaruudellista tasaista jakaumaa silmällä pitäen pistepilvien näytteistys suoritetaan vokselihilasuotimella. Vokselihilasuodin toteutettiin itse, sillä PCL:n molemmat vokselihilasuotimet, **UniformSampling** ja **VoxelGrid**, havaittiin toimimattomaksi käyttötarkoitukseen. Kummankin toteutuksessa käytetään vokseleiden indeksinä vain 16-bittiseksi taattua ja yleisesti 32-bittistä kokonaislukumuuttujaa. Tällöin pistepilven realistisella kymmenienkin metrien ulottuvuuskohtaisella sivumitalla ja järkevän pienellä vokselikoolla mahdolliset indeksin arvot vuotavat helposti yli muuttujan suurimmasta esitettävissä olevasta arvosta. Ongelma ratkeaa helposti käyttämällä 64-bittistä kokonaislukumuuttujaa, joka onkin tärkein ero toteutetun suotimen ja **UniformSampling**:n välillä. Algoritmisesti ne ovat yhtenevät – kunkin vokselin kattamalta alueelta suotimen läpäisee vokselin keskipistettä lähin pistepilven piste.

Ideallisesti vokselikoon pitäisi olla pienempi kuin mitattavan betonikerroksen paksuus, sillä tällöin vastaavuuksien maksimietäisyytenä voisi helpommin käyttää niin pientä arvoa, että muuttunut betonoitu pinta jäisi huomioimatta kohdistettaessa. Vokselikoon valinnalla on kuitenkin suora vaikutus suodatettujen pilvien pistetiheyteen ja näin ollen kohdistusalgoritmin ajoaikaan, joten liian pienikään arvo ei voi olla. Näiden ristiriitaisten vaatimusten kompromissina ohjelmassa käytetään vokselikokona sivumitaltaan 10 cm kuutiota.

Naapurustohakuja varten tarvittavana kiihdytysrakenteena käytetään PCL:n tarjoamaa **KdTree**:tä, joka delegoi KD-puun toteutuksensa FLANN-kirjastolle (Fast Library for Approximate Nearest Neighbours) [31]. Nimestään huolimatta FLANN kykenee myös eksaktiin naapurustohakuun perinteisellä KD-puulla, jota PCL tässä yhteydessä käyttää. FLANN ei ole tässä käyttötarkoituksessa tehokkain mahdollinen kirjasto [32], mutta ohjelmiston toimivuuden tai asymptoottisen  $\mathcal{O}$ -suorituskyvyn kannalta tällä ei tietenkään ole merkitystä. PCL käyttää KD-puun lehtisolmujen pistemääränä 15:tä ja FLANNin KD-puun solmujen jakotapa on pisimmän ulottuvuuden mediaanipisteessä jakava taso [31].

Normaalien laskenta pinnan approksimoinniksi suoritetaan ohjelmistossa piste-kohtaisesti PCL:n **NormalEstimation(OMP)**-algoritmilla, joka käyttää luvussa 4.4.3

esitettyä PCA-pohjaista tasosovitteen approksimointitapaa. Tasosovite tehdään pistejoukolle, joka on 25 cm sisällä käsiteltävästä pisteestä. Tällöin yhtenäisten pintojen pisteille pitäisi väistämättä löytyä naapuripisteitä vielä vokselihilasuodatuksen jälkeenkin. Lisäksi normaalien pitäisi käyttäytyä suhteellisen tasaisesti, mutta kuitenkin seurata pinnanmuotoja tarpeellisissa määrin; tunnelien seinämissä harvoin on merkittävää geometriaa pienemmässä kuin kymmenien senttimetrien suuruusluokassa etenkin kohdistamista ajatellen. Näin lasketut normaalit ovat käyttökelpoiset, vaikkakin ongelmallisemmat, myös pistepilvien erotuksen laskentaa varten, sillä ne voivat vääristää etäisyysmittaa terävien piirteiden kohdalla.

### 5.2.3 Skannien kohdistaminen

Referenssi- ja mittauspilvien kohdistus suoritetaan ICP-algoritmivariaatiolla, joka käyttää vastaavuuksien haussa raakaa lähimpien pisteiden etsintää. Vastaavuuksia ei periaatteessa karsita erikseen, vaan niitä etsiessä huomioidaan vain vastaavuudet, jotka ovat korkeintaan määrätyllä etäisyydellä pisteestä. Käytännössä PCL:n KD-puun toteutuksen rajoituksista johtuen tämäkin kuitenkin tehdään vasta varsinaisen lähimmän naapurin haun jälkeen, eikä jo sen sisällä. PCL:n KD-puun toteutuksen yleinen tehottomuus on myös suuri motivaatio raa'alle lähimpien pisteiden etsimiselle, sillä tällöin suoritetaan minimimäärä naapurustohakuja ja prototyypiohjelmiston suorituskyyky pysyy mahdollisimman vertailukelpoisella tasolla.

Virhemetriikkana ICP-algoritmissa käytetään vastaavuuksien pisteestä-tasoon-etäisyyksien neliösummaa, jonka minimointi suoritetaan linearisoimalla optimointiongelman olettamalla kohdistavan transformaation kulmavirhe pieneksi luvussa 4.6.4 esitetyn mukaisesti. Kirjallisuudessa tätä pidetään yleisesti tehokkaimpana yhdistelmänä konvergoitumisnopeuden ja monesti myös lopputuloksen tarkkuuden kannalta, eivätkä prototyypiohjelman kehitysvaiheessa tehdyt havainnot ole tämän kanssa ristiriidassa. Pisteestä-pisteeseen-etäisyyttä kokeiltaessa ICP-algoritmi konvergoitui kauttaaltaan hitaammin etenkin, kun kohdistavan transformaation alkuarvo oli suhteellisen huono.

Huonoimpien vastaavuuksien vaikutuksen vähentämiseksi lopullisen kohdistuksen tarkkuuteen ICP ajetaan kahdessa osassa siten, että ensimmäisellä kerralla käytetään suurempaa, yhden metrin, vastaavuuksien sallittua maksimietäisyyttä. Tällä maksimietäisyydellä lopputulos ei ole välttämättä kovin tarkka, koska ilmiselvästi ja pahasti vääriä vastaavuuksia syntyy väistämättä lukuisia. Se kuitenkin auttaa pääsemään parempaan kohdistuneisuuteen seuraavalle ICP:n ajokerralle, jolla vastaavuuksien maksimietäisyytenä käytetään tarkemman lopputuloksen saavuttamiseksi 10 cm:ä. Esimerkiksi Pulli [52] esittää käytettäväksi samankaltaista laskevaa maksimietäisyyttä, joskin hivenen pidemmälle vietynä. Huomattavasti 10 cm maksimietäisyys ei suinkaan poissulje betonoidun alueen pisteiden käyttämistä kohdistuksen vastaavuuksina. Kohdistettavien pilvien pistetiheyden täytyisi olla huomattavasti suurempi kuin käytetyllä vokseloinnilla saavutetaan, jotta maksimietäisyyttä voisi laskea riittävästi näiden pisteiden poissulkemiseksi.

### 5.2.4 Betonikerroksen paksuuden määrittäminen ja visualisointi

Toteutetussa ohjelmassa lasketaan skannien erotus pisteestä-tasoon-määritelmällä vokselihilalla näytteistetyistä pistepilvistä. Valinnan huonona puolena rosoisimmissa kohdissa etäisyysmitta on väistämättä pienempi kuin kerroksen todellinen paksuus. ICP:n löytämän kohdistavan transformaation avulla olisi mahdollista myös laskea etäisyys vokseloimattomien pistepilvien välillä tai vaihtoehtoisesti vokseloidun mitauspilven ja vokseloimattoman referenssipilven välillä. Valittu ratkaisu on kuitenkin yksinkertaisin ja tehokkain, sillä se välttää tarpeen ylimääräisille korkeamman pisteitiheyden kiihdytysrakenteille.

Pistekohtaisesti laskettu mittaus- ja referenssipilvien erotus esitetään käyttäjälle yksinkertaisuussyistä yhtenä lämpökarttamaisesti värjättyinä pistepilvenä, jossa punainen vastaa muuttumatonta kohtaa seinämässä, vihreä ruiskubetonikerroksen tavoitepaksuutta ja siniset sävyt tavoitepaksuuden ylittävää betonikerrosta. Teknisesti ottaen mittauspistepilveä rikastetaan lasketulla pistekohtaisella tiedolla etäisyydestä referenssipilveen. Tämä skalaarinen etäisyys muunnetaan väriarvoksi HSV-väriavaruudessa (sävy, kylläisyys ja kirkkaus; engl. hue, saturation and value) siirtämällä ja skaalaamalla arvot asetetulta etäisyysväliltä sävyarvovälille  $[0^\circ, 240^\circ]$  eli väreiksi punaisesta vihreän kautta siniseen vakiolla kylläisyydellä ja kirkkautella. HSV-väriavaruutta käytetään sen yksinkertaisuuden vuoksi; ihmisen värien havainnointikyvyille lineaariselta käytökseltään parempia, joskin mutkikkaampia, väriavaruuksia on olemassa [64]. Värjätyn pistepilven visualisointi suoritetaan PCL:n tarjoamalla `PCLVisualizer`-avustimella, joka piirtää pistepilven kolmiulotteisena ja mahdollistaa sen vapaan tarkastelun eri sijainneista, kulmista sekä projektioista.

## 5.3 Havainnot ja toimivuus

Toteutettua prototyyppijärjestelmää testattiin autenttisessa tunnelityömaaympäristössä skannien ottamisen ja ajoituksen osalta. Varsinainen laskenta tehtiin käytännöllisyyssyistä jälkikäteen. Koska kyseessä oli aktiivinen tunnelityömaa, saatiin ensikäden tietoa järjestelmän toimivuudesta suunnitellussa ympäristössä. Käsitellään seuraavaksi mitausjärjestelmän toiminnasta tehtyjä havaintoja ja analysoidaan niitä.

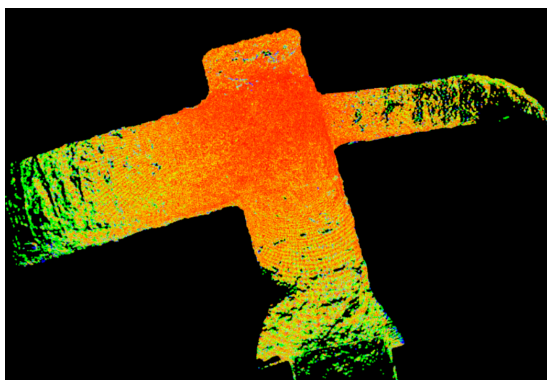
### 5.3.1 Skannien kohdistuminen

Luvussa 3.4 esitetyn mukaisesti prototyypin yleinen toimivuus sekä tulosten hyvyys riippuu pitkälti referenssi- ja mitausskannien kohdistamisen onnistumisesta. Skannien välisen eroavaisuuden ja toisaalta työmaalle aiheutuvan häiriön minimoimiseksi skannaaminen oli suoritettava ruiskutuslaitteiston odottaessa seuraavan betonikuorman saapumista eli potentiaalisesti alle 10 minuutissa. Tässä ajassa kerkeää hyvin pystyttää laserskannerin kolmijalalle ja ottaa skannin, mutta sensorin täsmällisen sijainti- ja orientaatiotiedon hankkimiseen esimerkiksi takymetrillä aika ei olisi mitenkään riittänyt. Suoraa tietoa oikeasta kohdistumisesta ei siis ole ja kohdistumisen hyvyttä on mahdotonta arvioida eksaktisti, mutta hyviä arvioita on mahdollista esittää kohdistuksen lopputuloksen pohjalta. Käytännössä skannit vaikuttavat

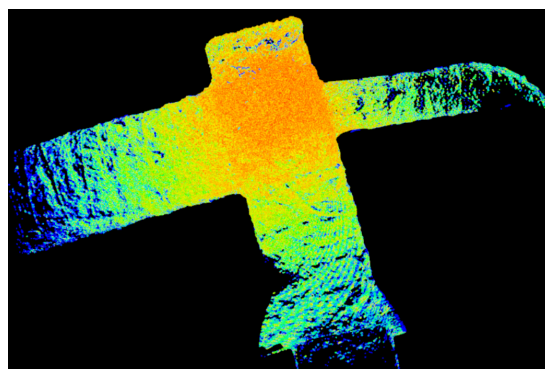


kohdistuvan hyvin, sillä silmämääräisesti skannien reuna-alueidenkin pintojen todettiin olevan hyvin kohdistuneet, ja pistepilvien välinen laskennallinen pistekohtainen etäisyys on verrattain pieni.

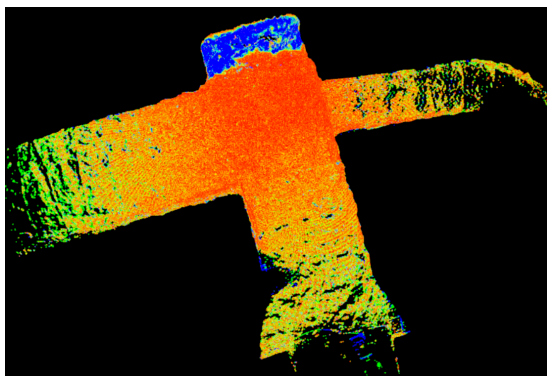
Kohdistumisen hyvyyden arvioimiseksi kohdistettujen pistepilvien pistekohtaista etäisyyttä mitattiin kohdistetun 10 cm -sivumitalla vokseloidun mittauspilven sekä 1 cm -sivumitalla vokseloidun referenssipilven välillä. Tällöin suoritusaika on kohtalaisen paljon suurempi kuin 10 cm -näytteistettyä referenssipilveä käyttäessä, mutta tulos antaa tarkemmin kuvan kohdistumisen onnistumisesta. Kohdistettujen mittausta- ja referenssipilvien välinen etäisyys eräissä tapauksissa on esitetty kuvassa 12. Voidaan nähdä, että tyypillisesti suunnitellulla mittausetäisyydellä lähellä



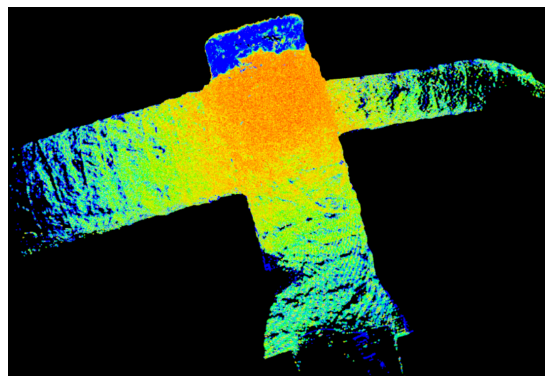
(a) Skannit otettu yhtenevästä tilanteesta läheltä toisiaan. Etäisyys mitattu pisteestä-tasoon-menetelmällä.



(b) Skannit otettu yhtenevästä tilanteesta läheltä toisiaan. Pisteestä-pisteeseen-etäisyys on konsistentisti pisteestä-tasoon-etäisyyttä suurempi kuten ennustettua.



(c) Skannit otettu ennen ja jälkeen ruiskubetonoinnin. Etäisyys mitattu pisteestä-tasoon-menetelmällä. Betoineitu kohta erottuu selvästi.



(d) Skannit otettu ennen ja jälkeen ruiskubetonoinnin. Pisteestä-pisteeseen-etäisyysmitta käyttäytyy konsistentisti muiden kombinaatioiden kanssa.

Kuva 12: Kohdistettujen skanniparien pistekohtaisia etäisyyksiä eri etäisyydsmäärittelyillä muuttumattomassa ja ruiskubetonoinnin muuttamassa tilanteessa. Etäisyydet ilmaistu värein punaisesta (0 mm) vihreän (15 mm) kautta siniseen (>30 mm). Skanneri sijaitsee kussakin kuvassa hivenen tunnelin risteämän keskikohdan yläpuolella.

Taulukko 1: Kohdistettujen skannien pistekohtaisten etäisyyksien tunnuslukuja.

	Pisteestä–tasoon		Pisteestä–pisteeseen	
	Keskiarvo	Mediaani	Keskiarvo	Mediaani
<b>Yhtenevä skannipari</b>	17 mm	2,9 mm	33 mm	8,8 mm
<b>Eroava skannipari</b>	19 mm	3,1 mm	46 mm	9,5 mm

skanneria valtaosa pisteistä on pisteestä–tasoon-määritelmän mukaisesti siirtynyt skannien välillä 0–5 mm ja kauempanakin skannerista 0–15 mm. Vastaavat pisteestä–pisteeseen-etäisyydet taas ovat noin 0–7 mm ja 0–22 mm. Mitattu virhe sekä sen kohina ovat suuremmat kuin skannerin pitäisi aiheuttaa. Todennäköisesti kohdistumisessa on siis systemaattinen virhe, jonka aiheuttaa luultavimmin skannin ulkolaidoilla harveneva pistedata.

Huomattavasti kohdistumistarkkuudessa on hyvin vähän eroa muuttumattomien ja ruiskubetonin osalta muuttuneiden skanniparien välillä. Kuvaparien 12c ja 12a sekä 12d ja 12b välillä (sivulla 38) on vaikea silmin havaita eroa. Laskennallisesti havaitaan, että muun kuin skannerin eksaktin sijainnin osalta yhtenevien skanniparien sekä ennen ja jälkeen ruiskubetonoinnin otetun skanniparin pistekohtaisten etäisyyksien jakaumassa on eroja, jotka on esitetty taulukossa 1 keskiarvon ja mediaanin avulla. Keskiluvuissa näkyvän 0,2–13 mm nousun pohjalta on epävarmaa, onko kohdistustulos huonontunut muuttuneiden skannien välillä, sillä betonoidun kohdan muutos skannissa pitääkin näkyä etäisyyksien keskilukujen nousuna. Karkeasti arvioiden 13 mm muutos yhtenevien ja eroavien skanniparien välillä raa’alla pisteestä–pisteeseen-etäisyydellä on kuitenkin enemmän kuin yhtä hyvälle kohdistumiselle olisi luonnollista. Betonoinnin takia muuttunut osa skannista vastaa kuitenkin vain noin kahdettakymmenettä osaa sen pinta-alasta ja kyseisessä mittaustapauksessa betonikerroksen paksuus on karkeasti 10 cm. Muutkin ympäristön muutokset kuten tunnelissa toimivien ajoneuvojen siirtyminen skannien välillä kuitenkin vaikuttavat väistämättä tuloksiin, samoin kuin erilainen skannien reuna-alueiden hajapisteiden sijainti. Voitaneen päätellä, että betonoitu seinämän osa ei ole skannien kohdistumisen määräävä virhelähde, vaikka se todennäköisesti jossain määrin vaikuttaakin.

Kohdistumisen epäideaalisuudesta huolimatta lähellä skanneria kohdistumisen virhe on suhteellisen pieni. Realistisena viitearvona tälle betonikerroksen paksuuden mittausrvirheelle voidaan kuvan 12 edellisellä sivulla silmämääräisen tarkastelun perusteella pitää kohdistettujen skannien pistekohtaisten etäisyyksien mediaania, joka on taulukon 1 mukaisesti pisteestä–tasoon-etäisyydellä suuruusluokaltaan 3 mm ja pisteestä–pisteeseen-etäisyydellä 10 mm. Huomattavasti etäisyyksien mittaaminen 1 cm sivupituudella vokseloituun referenssipilveen aiheuttaa yhtenäiselläkin pinnalla pahimmillaan  $\frac{5\sqrt{3}}{2}$  mm eli 4,3 mm positiivisen vääristymän pisteestä–pisteeseen-etäisyyteen. Vääristymän suuruus tosin laskee pintojen loitotessa toisistaan. Toisaalta luvussa 4.7 mainitun mukaan pisteestä–tasoon-etäisyys on aina sama tai pienempi kuin todellinen pintojen välinen etäisyys. Varovaisesti arvioiden kohdistumisen mahdollistama paras saavutettava betonikerroksen paksuuden mittausrvirhe suunnitellulla etäisyydellä on siis suurella todennäköisyydellä pisteestä–tasoon- ja

pisteestä–pisteeseen-etäisyysmittojen mediaanien välillä, eli noin 6 mm. Käytännössä saavutettava mittausvirhe lienee merkittävästi pienempi, mutta tätä on hyvin vaikea taata.

Merkittävä tekijä kohdistumisen tarkkuuden kannalta on käytetyn algoritmiikan lisäksi myös syötedata ja on tärkeää huomata kuvassa 12 sivulla 38 esitellyn sijainnin tunnelissa olevan kohdistamisen kannalta helppo, sillä siinä on selkeitä isoskaalaisia piirteitä. Kyseisessä tapauksessa toteutetun ohjelmiston todettiin mittauspilveä jäykästi transformoimalla kykenevän kohdistamaan skannit karkeasti 2 m translaatio- ja 20° rotaatiovirheestä, mikä on käyttötapauksessa varsin riittävä kovergenssialue. On kuitenkin melko varmaa, että vaikeampien tapausten kanssa, kuten pitkää suoraa ja yhtenäistä tunnelia skannatessa kovergenssialue on huomattavasti pienempi, mahdollisesti jopa käyttökelvottoman pieni. Tämä on ICP-algoritmin yksinkertaisen toimintamekanismin varjopuoli, jota on vaikea kiertää ilman kehittyneempiä kohdistusmenetelmiä [16].

### 5.3.2 Betonikerroksen paksuus

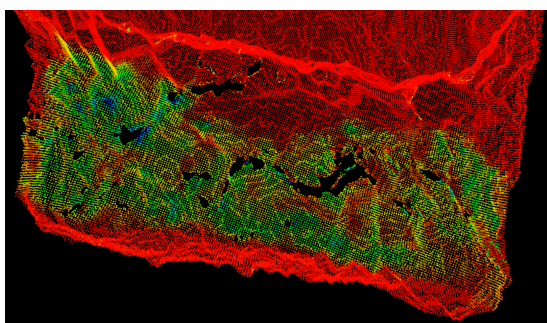
Kohdistamisen jälkeen toiseksi kriittisin osa prototyypijärjestelmää on itse betonikerroksen paksuuden mittaaminen kohdistetuista pilvistä. Mitatun paksuuden oikeellisuutta ei valitettavasti voitu ehdottomasti todentaa, koska se olisi vaatinut runsasta tuhoavaa näytteenottoa työnvaiheessa, jossa siitä olisi ollut työmaan etenemiselle olennaista haittaa. Arvioidaan siis mitatun paksuuden oikeellisuutta pohjautuen silmämääräiseen tarkasteluun sekä ruiskuttajan itsensä esittämään tavoiteltuun kerrospaksuuteen ja havaintoihin ruiskutuksen etenemisestä.

Esimerkkikäyttötapauksessa tarkoituksena oli ruiskuttaa tunnelin pohjukan holviosaan eli sivuseinien ja katon paljaalle kalliopinnalle 10 cm kerros ruiskubetonia. Ruiskutuksen aikana useammasta kohdasta kattoon tippui useamman litran kokoisia betonikerroksen paloja omamassansa vaikutuksesta. Näiden tietojen pohjalta voidaan arvioida mittaustuloksen oikeellisuutta. Myös päätelmiä mittaustuloksen yleisestä käyttökelpoisuudesta voidaan tehdä tältä pohjalta.

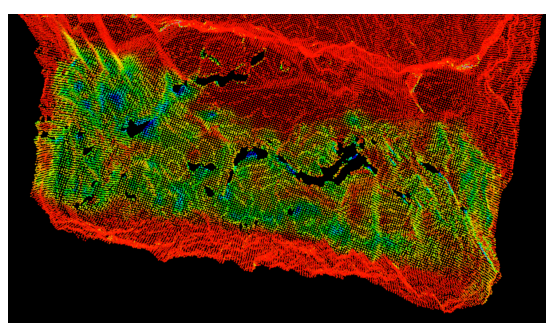
Asettamalla käytetyt kohdistettujen pistepilvien pistekohtaisten etäisyyksien värit oikeiksi esimerkkitapauksen suhteen eli kuvaamalla 0 cm etäisyys punaisella, tavoitepaksuus 10 cm vihreällä ja tämän ylittävät etäisyydet 20 cm:ssä saturoituvalla sinisellä, saadaan kuvan 13 mukaiset mittaustulokset seuraavalla sivulla. Eri etäisyysmitoilla sekä referenssiskannin vokseliresoluutiolla on laskettuun lopputulokseen vaikutusta, mutta käytännössä pisteestä–tasoon-etäisyyttä käyttäessä tulos on yhtä hyvä sekä 100 mm että 10 mm -vokseloituun referenssiskanniin mitaten. Myös pisteestä–pisteeseen-etäisyys 10 mm -vokseloituun referenssiskanniin on hyvin samankaltainen edellisten kanssa. Vain pisteestä–pisteeseen-etäisyys 100 mm -vokseloituun referenssiskanniin eroaa merkittävästi muista, mutta sekin enimmäkseen muuttumattomien ja vähän muuttuneiden alueiden osalta. Vähän muuttuneilla alueilla etäisyysmitta vääristyy ison vokselikoon vaikutuksesta pahimmassa tapauksessa  $\frac{50\sqrt{3}}{2}$  mm eli 43 mm, joka onkin lähellä 10 cm mittaustuloksen yleistä ”nollatasoa”; 50 mm vastaa värien määrittelyn mukaan kirkasta keltaista. Kaiken kaikkiaan mittaustulos on hyvin johdonmukainen eri etäisyysmittojen sekä referenssiskannin resoluutioiden

den suhteen. Prototyypiohjelmistossa käytetty 10 cm -vokseloitu referenssipilvi pisteestä-tasoon-etäisyysmitalla kuvassa 13c tuottaa siis vaihtoehtojaan vastaavan mittaustarkkuuden, paremman tai yhtä hyvän suorituskyvyn sekä vääristymättömän kuvan muuttumattomista alueista.

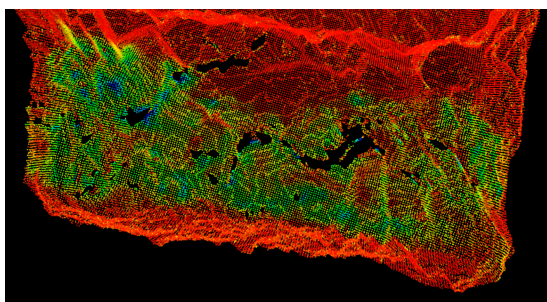
Suurempiresoluutioisiin vaihtoehtoihin suhteutetun luotettavuutensa lisäksi prototyypiohjelmiston tuottama tulos on konsistentti ruiskutusta seurattaessa tehtyjen havaintojen sekä ruiskuttajan omaan kerrospaksuustavoitteen ja -arvion kanssa. Valtaosalla betonoitua aluetta kerrospaksuus on hyvin lähellä tavoiteltua 10 cm:ä tai vaihtoehtoisesti 0 cm:ä; mitatut ”aukot” sijaitsevat pitkälti kohdissa, joista betonია tippui ruiskutettaessa liiallisen kerrospaksuuden aiheuttaman massan johdosta. Tavoitekerrosta pienempiä paksuuksia on kuitenkin myös muilla kuin edellä mainituilla alueilla, etenkin kuvassa 13 esitetyn näkymän oikean laidan seinämän ja katon välillä. On vaikea sanoa, onko kyseessä mittausvirhe vai epähuomiossa tavoitetta ohuempaksi jäänyt betonikerros, mutta esitetyt havainnot puoltavat tavoitetta ohuempaa betonikerrosta.



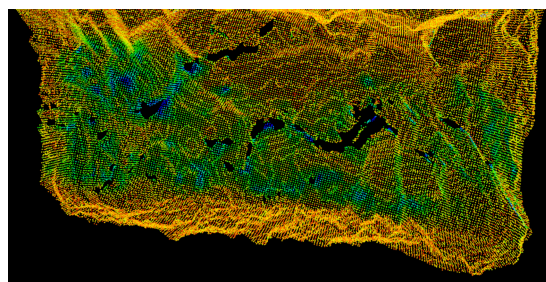
(a) Etäisyys mitattu pisteestä tasoon 1 cm -vokseloituun referenssiskanniin.



(b) Etäisyys mitattu pisteestä pisteeseen 1 cm -vokseloituun referenssiskanniin.



(c) Etäisyys mitattu pisteestä tasoon kohdistuksessa käytettyjen 10 cm -vokseloitujen skannien välillä. Vastaa luvussa 5.2 esitettyä ohjelmistoratkaisua.



(d) Etäisyys mitattu pisteestä pisteeseen kohdistettujen pilvien välillä. Etäisyysmitta vääristää muuttumatonta aluetta pahasti, mutta muuttunut alue on konsistentti vertailukohtien kanssa.

Kuva 13: Mitattu ruiskubetonikerroksen paksuus eräässä tapauksessa ilmaistuna värein punaisesta (0 cm) vihreän (10 cm) kautta siniseen (>20 cm). Tunnelin pohja leikattu pois näkyvistä, näkökulma alhaalta ylöspäin.

Mitatun paksuuden konsistenttiudesta huolimatta on huomattavaa, että kohdistuksen epätäydellisestä aiheutuu tulokseen melko varmasti jonkinlaista systemaattista virhettä. Ei ole kuitenkaan erityistä syytä epäillä, etteikö mitattu paksuus etenkin tasaisissa kohdissa skannattua aluetta olisi kohdistuksen salliman mittausvirheen eli karkeasti noin 6 mm päässä oikeasta betonikerroksen paksuudesta. Edes olettaen pessimistisesti muuttuneiden pintojen imeneen toisiaan puoleensa niin paljon kuin kohdistuksen tarkkuuden muutoksen keskiluvut (ks. taulukko 1 sivulla 39) sallivat, on vaikea uskoa, että kokonaismittausvirhe nousisi 10 mm:ä suuremmaksi. Mitattu paksuus on myös käytännössä varmasti pienempi kuin todellinen, sillä sekä kohdistus että etäisyyden laskennassa käytetty pisteestä–tasoon-etäisyys vääristävät tulosta tähän suuntaan.

Kaiken kaikkiaan prototyyppijärjestelmällä ei siis ole mahdollista mitata millintarkasti ja luotettavasti ruiskutetun betonikerroksen paksuutta, mutta senttimetrin suuruusluokassa tarkkuus kuitenkin erittäin todennäköisesti on. Mitattu kerrospaksuus on näin ollen käyttökelpoinen paikallistamaan alueita, joilla on mahdollisesti turhan paksu tai ohut betonikerros ja puuttumaan asiaan halutulla tavalla. Etenkin kerrokseltaan liian ohuen alueen pitäisi mittausvirheen todennäköisen suunnan ansiosta erottua vääjäämättä. Poikkeuksen mitatun paksuuden luotettavuuteen tuonevat vain todella terävät pinnanmuodot sekä sensorin katveet. Näistä ensimmäistä voi edesauttaa helposti suorittamalla laskennan näin vaadittaessa korkeammalla tarkkuudella ja jälkimmäinen on sensoriteknikasta aiheutuva vääjäämätön rajoite, jonka ainoa ratkaisu on koittaa saada sensori mahdollisimman lähelle ruiskutettua pintaa.

### 5.3.3 Käyttökelpoisuus työmaaoloissa

Ruiskubetonin kerrospaksuuden mittaamisen itsensä lisäksi luvussa 3.3 visioidun kokonaismittausjärjestelmän käyttökelpoisuuteen vaikuttaa myös monet muut asiat. Näistä tärkein lienee järjestelmän käyttökelpoisuus työmaaolosuhteissa, sillä tieto betonikerroksen paksuudesta olisi hyödyllistä saada nopeasti käyttöön nimenomaan työmaalla, jossa sensoridatakin tuotetaan joka tapauksessa. Prototyypin ohjelmistoa ei testattu käytännöllisyysyistä tunnelityömaaoloissa, mutta skannit otettiin autenttisessa ympäristössä työmaan aikataulua täysin kunnioittaen.

Eräs tärkeä mittausjärjestelmän käyttökelpoisuustekijä on luvussa 2.3 mainitun mukaisesti skannaamiseen, tiedon siirtämiseen sekä suoritettavaan laskentaan kuluva aika. Kerrospaksuustiedon ollessa saatavilla ennen betonoinnin jatkumista mahdolliset virheet voidaan korjata heti. Toteutetulla ohjelmistolla ja käytetyillä skannerin asetuksilla aikaa eri mittausjärjestelmän osissa kuluu seuraavasti:

- Skannaaminen n. 3 min skannia kohden
- Tiedonsiirto skannerilta tietokoneelle n. 30 s skannia kohden
- Kaikki skannien prosessointi yhteensä n. 30 s.

Prototyyppiä kehittäessä ja testatessa kaikki skannaamisen jälkeiset vaiheet suoritettiin jälkikäteen, mutta valmiissa järjestelmässä referenssiskannin tiedonsiirto ja osa

sen prosessointia on mahdollista tehdä jo ruiskubetonoinnin aikana. Periaatteessa aikaa ruiskutuksen lopettamisesta kerrospaksuuden laskennan valmistumiseen ja visualisointiin kuluu siis skannaamiseen ja tiedonsiirtoon kuluva aika kertaalleen sekä prosessointiin kuluva aika, eli noin 4 min. Visioidussa ”valmiissa” järjestelmässä kuluisi lisäksi jonkin verran aikaa ruiskutuslaitteen robottikäsiarven ajamiseen sopivaan asentoon ja skannerin suojan avaamiseen, arviolta noin 30 s. Toisaalta näytteistyksen harvuuden johdosta skannatessa voisi sangen hyvin käyttää myös matalampaa kulmaresoluutiota, jolloin käytetyn skannerin tapauksessa yhteen skanniin kuluva aika puolittuisi noin 1,5 minuuttiin ja tiedonsiirtoonkin kuluisi neljännes alkuperäisesti ajasta. Näin siis saavutettavissa oleva viive visioidussa järjestelmässä ruiskutuksen lopettamisesta betonikerroksen paksuuden näkemiseen olisi karkeasti kolme minuuttia. Tämä on käyttökelpoinen aika, sillä ruiskubetonointiprosessin taukojen havaittiin työmaalla yleisesti olevan ainakin kymmenenminuuttisia. Mittaustulokseen siis keretään haluttaessa reagoida heti kun ruiskutus käynnistyy uudestaan.

Betonikerrospaksuuden mittaamiseen kuluvan ajan lisäksi työmaakäytön kannalta merkittävää on instrumentointiratkaisun yleinen toimivuus tunnelityömaaolosuhteissa. Prototyypin mittauksia tehdessä ei havaittu olennaisia ongelmia visioidun kokonaisjärjestelmän kannalta, sillä ruiskutustyön tauotessa luonnollisesti esimerkiksi käytetyn betonikuorman vaihtuessa on hyvin aikaa ja tilaa ottaa skanni vastabetonoidusta kohdasta tunnelia; betonoidussa kohdassa tunnelia ei ole joka tapauksessa liikennettä ja kuten mainittua tarvittavan ajan ei pitäisi olla ongelma edes prototyypissä saavutetulla suorituskyvyllä. Tila- ja aikarajoitteiden lisäksi tunnelityömaan olennaisin häirttekijä on kosteus, joka ilmenee tunnelin pohjan pehmeiden ja lamikkaisuuden lisäksi myös seinämien vaihtelevana kosteutena. Kohdistustarkkuudesta ja betonikerroksen paksuuslaskennasta tehtyjen havaintojen pohjalta kosteus ei kuitenkaan merkittävästi vaikuta tuloksiin ja niin kolmijalalla kuin robottikäsiarrellekin tuettuna skannerilaitteisto on hyvin pahimmalta kosteudelta suojassa.

## 5.4 Jatkokehitysmahdollisuudet

Prototyypin toiminnasta ja tunneliolosuhteista tehtyjen havaintojen pohjalta sekä prototyypin itsensä että sen koestaman visioidun kokonaisjärjestelmän kannalta järjestelmässä on monia paranneltavissa olevia yksityiskohtia. Ilmeisimpiä näistä ovat kohdistumisen tarkkuus sekä varmuus piirreköyhemmissä tunnelin osissa. Myös järjestelmän toimivuus ilman asiantuntijakäyttäjää sekä siirtyminen prototyypistä käyttökelpoiseksi mittausjärjestelmäksi vaatisivat runsaasti lisäkehitystä.

Kohdistumisen tarkkuutta on todennäköisesti mahdollista parantaa suodattamalla skanneista hajapisteet ja -pistejoukot paremmin. Lidarin mittausvalomäärään pohjautuva suodatus ja vastaavuuksien maksimietäisyys vähentävät näiden vaikutusta olennaisesti, mutta suodatusta pystyisi tekemään paremmin tässä työssä esitellyn menetelminkin. Mahdollinen ongelma lisäsuodatuksessa on sen suorituskykyvaikutus. Tällä hetkellä prototyyppijärjestelmän mittaus–visualisaatio-syklissä kuluu kuitenkin skannaamisen moninkertainen määrä aika skanniin ohjelmalliseen prosessointiin verrattuna. Ohjelmiston lievää hidastumista paremman kohdistustarkkuuden saavuttamiseksi on vaikea pitää ylitsepääsemättömänä ongelmana.

Myös skanneista saatavien pistepilvien, etenkin referenssipilven, tiheämmällä vokseloinnilla tai muulla näytteistyksellä lieneisi mahdollista parantaa kohdistusta vastaavuuksien maksimietäisyyden avulla; jos vastaavuusetäisyyden saa laskettua alle betonikerroksen odotetun paksuuden, betonoidun pinnan ei pitäisi juuri vaikuttaa kohdistumiseen. Käytännössä tämä havaittiin ilman ohjelmiston runsasta optimointia liian hitaaksi vaihtoehdoksi, mutta PCL-kirjaston turhia suorituskykypullonkauloja korjaamalla tämä olisi todennäköisesti järkevä vaihtoehto ilman järeämpiä optimointeja kuten näytönohjainpohjaista laskentaa. KD-puu-toteutuksensa hitauden lisäksi PCL:a hidastaa sen jähmeä ohjelmointirajapinta, joka tekee tarpeettoman työlääksi rakentaa pistepilvelle KD-puu, laskea sen kiihdyttämänä normaalit ja sitten käyttää uudelleen jo rakennettua KD-puuta normaalein rikastetun pistepilven kanssa. Prototyypiohjelmistossa tämä jähmeys ratkaistiin erittäin epäoptimaalisesti rakentamalla KD-puu kahteen kertaan pistepilveä kohden, ensin ilman normaaleja, sitten normaalien kanssa.

Suorituskykyparannuksilla saatavan raa'an lisälaskentavoiman salliman korkeamman resoluution ja paremman suodatuksen lisäksi ICP-algoritmin kohdistustarkkuutta ja ennen kaikkea varmuutta piirreköyhemmissä tunnelin osissa voi olla vaikea parantaa. Tätä varten voisi olla hyödyllistä käyttää avustavaa karkeakohdistusmenetelmää tai jotakin muuta hienokohdistusmenetelmää. Karkeakohdistusmenetelmän käyttämisen ylimääräinen hyvä puoli olisi, että se sallisi ruiskutusalueen mukana kulkevan skannerin liikkua skannien välillä enemmän ilman käyttäjältä vaadittavaa syötettä. Todennäköisesti karkeakohdistus veisi kuitenkin suhteessa paljon aikaa eikä karkeakohdistuksessa yleisesti käytettyjen korkeaulotteisten ominaisuustietojen yksilöivä määrittely piirreköyhältä alueelta välttämättä ole sekään helppoa – kokeilematta tätä on tietenkin vaikea arvioida. Muista hienokohdistusmenetelmistä ainakin NDT on erittäin kiinnostava vaihtoehto, sillä sen kolmi- tai useampiulotteisen version käyttämisestä on tutkittu tunneleista laserskannattujen pistepilvien rekisteröintiongelman parissa [16]. Tulokset vaikuttavat hyvin lupaavilta ja suurimpia kysymyksiä NDT:n suhteen ovat kohdistumisen riittävä tarkkuus ja suorituskyky; tutkimusten ensisijainen fokus ei ole ollut senttimetrisuuruusluokan kohdistustarkkuuden saavuttaminen kohtuullisessa ajassa.

Mittaustuloksen tarkkuuden ja skannien kohdistamisvarmuuden lisäksi jatkokehittävää mittausohjelmistosta löytyy etenkin visualisoinnista, joka on tällä hetkellä työmaaolosuhteisiin lähellä käyttökelpotonta. Pistepilvien pyörittely näytöllä voi olla kosteissa ja kiireisissä tunnelioloissa suojavarusteisiin pukeutuneena turhauttavan työlästä. Paras mietitty ratkaisu visualisoinnin ongelmiin olisi koko tunnelin katon ja seinämien projisointi tasoon tunnelin suunnitellun profiilin mukaisesti tai automaattisesti. Näistä ensimmäisen ongelma on käyttäjältä tarvittava lisäsyöte, joka hankaloittaa järjestelmän käyttämistä. Toisaalta automaattinen tunnelin muodon havaitseminen ja käytävien levittäminen tasoon paikallisella projektiolla voi olla ongelmana hankala ratkaista yleisesti ja siististi.

Prototyypijärjestelmän puutteista ja epävarmuuksista huolimatta mikään siitä käyttäessä tehty havainto ei tee luvussa 3.3 visioidusta mittausjärjestelmästä mahdotonta saavuttaa. Ehkä suurin epävarmuustekijä mittausjärjestelmän yleisen käyttökelpoisuuden kannalta on toiminta piirreköyhemmillä tunnelialueilla. Tämän

ja muiden mainittujen kehitystarpeiden ja -mahdollisuuksien lisäksi järjestelmän kehittäminen “valmiiksi” edellyttää luonnollisesti isoa panostusta mekaaniseen ja sähkötekhniseen suunnitteluun, mutta ylitsepääsemättömiä esteitä hyvään lopputulokseen pääsemiselle on vaikea nähdä. Mielenkiintoisena lisänä ruiskubetonin kerrospaksuuden mittaamisen lisäksi ennen ja jälkeen betonoinnin otetuilla kohdistetuiden laserskannien pohjalta voi tehdä muutakin laskentaa. Ainakin seinämään ruiskuttaessa tarttuneiden betonikuutiometrien määrä lienee laadunhallinnallisesti mielenkiintoinen metriikka.



## 6 Johtopäätökset

Tunneliseinämien lujittaminen ruiskubetonoimalla on olennainen osa tunnelirakentamista niin työturvallisuuden kuin julkiseen käyttöön tarkoitettujen rakenteiden romahdusturvallisuudenkin kannalta. Nykyisin paksuutta valvotaan jälkikäteen poraten tehdyin hajamittauksin sekä työn aikana silmämääräisesti tai poraamista vastaavalla testimenetelmällä. Betonikerroksen kattavan paksuustiedon saaminen mahdollisimman nopeasti ruiskutuksen jälkeen lisäisi varmuutta kerrospaksuuden suunnitelmanmukaisuudesta. Tämä ehkäisisi tarvetta jälkikäteen tehtäville hankalille ja kalliille korjauksille sekä voisi auttaa vähentämään ylimääräisen, kustannusmielessä hukatun, betonin ruiskutusta.

Tunnelityömaa on toimintaympäristönä ja ruiskubetonointi työvaiheena erittäin haastava minkäänlaiselle instrumentoinnille tai tietotekniikalle. Kosteus, pöly ja kuivuneet pikkukiven kokoiset betoniroskeet sekä rajallinen tila hankaloittavat ruiskutusprosessin havainnointia. Tästä huolimatta instrumentointimahdollisuuksia betonikerroksen paksuuden seuraamiseksi on monia ja tässä työssä käsiteltiin niistä joitakin. Käyttökelpoisimmaksi vaihtoehdoksi esitettiin ruiskubetonialustan robottikäsiavarteen kiinnitettävää, hukkaroiskeilta suojattua 3D-laserskanneria. Tällä instrumentointiratkaisulla saadaan kattavia ja tarkkuudeltaan sekä datamäärältään erinomaisia syvyyskuva-aineistoja minimaalisilla katveilla häiritsemättä ruiskutusprosessia merkittävästi.

Runsaista hyvistä puolistaan huolimatta 3D-laserskanneri ei mittaa suoraan ruiskubetonikerroksen paksuutta, vaan se on laskettava pistekohtaisesti ennen ja jälkeen betonoinnin otettujen referenssi- ja mittausskannien erotuksesta. Betonoitu osa erottuu tällöin ainoana laajempaan yhtenäisenä alueena, jossa skannien välillä on muutosta. Tämän paksuudenmittaustavan huono puoli on vaatimus edellä mainittujen skannien erinomaisesta kohdistumisesta, joka on työmaaoloissa työlästä ja turhan hidasta toteuttaa fyysisin mittauksin esimerkiksi takymetrillä. Olennainen osa mittausjärjestelmää on referenssi- ja mittausskannien automaattinen kohdistaminen ohjelmallisesti. Skannien välisen pistekohtaisen etäisyyden eli betonikerroksen paksuuden laskeminen kohdistamisen jälkeen on suhteellisen vaivatonta.

Tämän työn puitteissa toteutettiin ehdotetun järjestelmän prototyypiversio, jossa käytettiin sensorina Leica HDS6200 -laserskanneria. Tarvittava mittausohjelmisto toteutettiin pitkälti pistepilvien käsittelyn apuvälineitä sisältävän PCL-ohjelmakirjaston avulla. Prototyypin toiminnasta tehtyjen havaintojen pohjalta esitetyn kaltainen mittausjärjestelmäkokonaisuus vaikuttaa toimivalta. Mittaustarkkuudeksi arvioidun noin 10 mm:n pohjalta ei voi etenkin ohuita betonikerroksia mitattaessa tehdä täsmällisiä johtopäätöksiä betonikerroksen paksuudesta, mutta pistekohtaisen paksuuden esittäminen värikarttamaisesti mahdollistaa silti keskimääräistä ruiskutettua kerrospaksuutta ohuempien tai paksumpien alueiden havaitsemisen. Myös tavoite kerrospaksuustiedon nopeasta saatavuudesta toteutui, sillä paksuustieto on prototyypijärjestelmän pohjalta arvioiden visualisoitavissa noin kolmen minuutin sisällä ruiskutuksen lopettamisesta ilman mittausohjelmiston optimoimistakin. Käytännössä ruiskubetonointi on aina tauolla ja tunneli skannattavissa pidempään kuin kolme minuuttia kerrallaan, joten tieto ruiskubetonikerroksen paksuudesta on

saatavilla ennen betonoinnin jatkamista. Betonikerroksen paksuusmittaus on siis ympäristö- ja prosessitekijät huomioiden riittävän reaaliaikainen, eikä vaadi työmaan prosessin ylimääräistä keskeyttämistä.

Prototyypin järjestelmän yleisestä toimivuudesta huolimatta sen toiminnassa havaittiin useita kehittämistä vaativia yksityiskohtia. Etenkin kohdistumistarkkuus voisi olla saavutettua parempi, vaikka periaatteessa jo prototyypin saavuttama tarkkuus onkin paksumpien kerrosten paksuuden mittaamiseen mahdollisesti riittävä sekä ennen kaikkea poikkeavien alueiden löytämiseen soveltuva. Myös oikean kohdistumisen saavuttamisen varmuus mittausjärjestelmällä etenkin piirreköyhissä, pitkissä ja suorissa tunneleissa jää avoimeksi kysymykseksi, sillä tällaista tapausta ei saatu testattua. Lisäksi on syytä muistaa, että työn puitteissa kehitetty prototyyppi on tarkoituksellisestikin kaukana valmiista järjestelmästä monilta osin, etenkin mekaanisen ja sähkösuunnittelun sekä kerrospaksuuden kätevän visualisaation puolesta. Vaikkei ylitsepääsemättömiä esteitä esitetyn kaltaisen mittausjärjestelmän toteuttamiselle havaittukaan, on toteutetusta prototyypistä vielä pitkä ja epävarma matka luotettavasti ilman asiantuntijakäyttäjää toimivaksi mittausjärjestelmäksi.

Kokonaisuutena tämä työ täyttää pääasiallisen tarkoituksensa. Sen puitteissa selvitettiin eräiden instrumentointiratkaisujen soveltuvuus lujiteruiskubetonin kerrospaksuuden työnaikaiseen seurantaan sekä esitettiin perustellusti soveltuvimmaksi ratkaisuksi ruiskubetonointilaitteistoon kiinnitettyä 3D-laserskanneria. Mikä tärkeintä, esitetyn instrumentointiratkaisun ja sen ympärille visioitun mittausjärjestelmän toimintaa testattiin prototyypin järjestelmällä. Tehtyjen havaintojen pohjalta voidaan todeta mittausmenetelmän olevan toimiva, instrumentointiratkaisun häiritsevän työmaan prosessia minimaalisesti ja saatujen mittauksien olevan rajoituksistaan huolimatta käyttökelpoisia. Kaiken kaikkiaan esitetty mittausjärjestelmä on jatkokehityskelpoinen ja toivottavasti jonain päivänä kehittyneemmässä muodossa ruiskubetonointityössä hyödyksi.

## Viitteet

- [1] Sprayed Concrete Association, *Introduction to Sprayed Concrete*, online [9.11.2017], 2005. url: [http://www.sca.org.uk/pdf\\_word/Intro\\_to\\_Sprayed\\_Concrete.pdf](http://www.sca.org.uk/pdf_word/Intro_to_Sprayed_Concrete.pdf).
- [2] A. Bergström, johtaja, Kalliorakentaminen ja kaivokset -liiketoimintayksikkö, Infrapalvelut, YIT Rakennus Oy, haastattelu, tammikuu 2016.
- [3] O. Schneider, M. Wehrli, J. Lehto ja I. Martikainen, “New innovative system for layer thickness control on Normet spray mobile”, teoksessa *ITA WTC 2015 Congress and 41st General Assembly*, Dubrovnik, Croatia, toukokuu 2015.
- [4] J. Lehto, *Ruiskubetonilaitteistot ja suutinmiehen pätevyys*, online [9.11.2017], 2015. url: [http://www.betoniyhdistys.fi/media/janne-lehto\\_ruiskubetonilaitteistot-ja-suutinmiehen-p-e4tevyys\\_01062015.pdf](http://www.betoniyhdistys.fi/media/janne-lehto_ruiskubetonilaitteistot-ja-suutinmiehen-p-e4tevyys_01062015.pdf).
- [5] S. Nabulsi, A. Rodriguez ja O. Rio, “Robotic Machine for High-Quality Shotcreting Process”, teoksessa *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, VDE, Munich, Germany, kesäkuu 2010, s. 1–8.
- [6] G Girmscheid ja S Moser, “Fully automated shotcrete robot for rock support”, *Computer-Aided Civil and Infrastructure Engineering*, vol. 16, nro 3, s. 200–215, 2001.
- [7] B. Simser, “Empirical experience with shotcrete in deep underground mines”, teoksessa *WSN Mining Health and Safety Conference*, 2014.
- [8] D. Kolymbas, “The New Austrian Tunnelling Method”, teoksessa *Tunnelling and Tunnel Mechanics: A Rational Approach to Tunnelling*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, s. 171–175, ISBN: 978-3-540-28500-7.
- [9] G Bernardo, A Guida ja I Mecca, “Advancements In Shotcrete Technology”, *WIT Transactions on The Built Environment*, vol. 153, s. 591–602, 2015.
- [10] Á. Rodríguez ja O. Río, “Analysis of real time technical data obtained while shotcreting: an approach towards automation”, teoksessa *ECCOMAS Thematic Conference on Computational Methods in Tunneling (EURO: TUN 2007)*, Vienna, Austria, elokuu 2007.
- [11] J. Salvi, S. Fernandez, T. Pribanic ja X. Llado, “A state of the art in structured light patterns for surface profilometry”, *Pattern recognition*, vol. 43, nro 8, s. 2666–2680, 2010.
- [12] M. J. Landau, B. Y. Choo ja P. A. Beling, “Simulating Kinect Infrared and Depth Images”, *IEEE Transactions on Cybernetics*, vol. PP, nro 99, s. 1–14, marraskuu 2015.
- [13] D. Fofi, T. Sliwa ja Y. Voisin, “A comparative survey on invisible structured light”, teoksessa *Electronic Imaging 2004*, International Society for Optics ja Photonics, 2004, s. 90–98.

- [14] M. Gupta, A. Agrawal, A. Veeraraghavan ja S. G. Narasimhan, “Structured light 3D scanning in the presence of global illumination”, teoksessa *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, 2011, s. 713–720.
- [15] C. English, S. Zhu, C. Smith, S. Ruel ja I. Christie, “Tridar: A hybrid sensor for exploiting the complimentary nature of triangulation and LIDAR technologies”, teoksessa *Proceedings of the 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2005.
- [16] M. Magnusson, “The three-dimensional normal-distributions transform: an efficient representation for registration, surface analysis, and loop detection”, tohtorinväitöskirja, Örebro universitet, 2009.
- [17] S. Fekete, M. Diederichs ja M. Lato, “Geotechnical and operational applications for 3-dimensional laser scanning in drill and blast tunnels”, *Tunnelling and Underground Space Technology*, vol. 25, nro 5, s. 614–628, 2010.
- [18] M. Reynolds, J. Doboš, L. Peel, T. Weyrich ja G. J. Brostow, “Capturing time-of-flight data with confidence”, teoksessa *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, 2011, s. 945–952.
- [19] J. Mure-Dubois ja H. Hügli, “Real-time scattering compensation for time-of-flight camera”, teoksessa *Proceedings of the ICVS Workshop on Camera Calibration Methods for Computer Vision Systems*, 2007.
- [20] H. M. Jol, *Ground penetrating radar theory and applications*. Amsterdam, Netherlands: Elsevier Science, 2008, ISBN: 978-0-08-095184-3.
- [21] G. Parkinson ja C. Ékes, “Ground penetrating radar evaluation of concrete tunnel linings”, teoksessa *12th International Conference on Ground Penetrating Radar. Birmingham, UK*, 2008, s. 11.
- [22] X Nunez-Nieto, M Solla, F. Prego ja H Lorenzo, “Assesing the applicability of GPR method for tunnelling inspection: Characterization and volumetric reconstruction”, teoksessa *Advanced Ground Penetrating Radar (IWAGPR), 2015 8th International Workshop on*, IEEE, 2015, s. 1–4.
- [23] F. Remondino, “From point cloud to surface: The modeling and visualization problem”, teoksessa *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XXXIV-5/W10, International Workshop on Visualization and Animation of Reality-based 3D Models, Tarasp-Vulpera, Engadin, Switzerland, helmikuu 2003.
- [24] R. B. Rusu ja S. Cousins, “3D is here: Point Cloud Library (PCL)”, teoksessa *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, toukokuu 2011.
- [25] H. Butler, B. Chambers, M. Gerlek ym., *PDAL: Point cloud Data Abstraction Library*, Release 1.5, online [25.7.2017], kesäkuu 2017. url: <http://pdal.io/PDAL.pdf>.

- [26] D. Girardeau-Montaut, *Cloud Compare User Manual*, Version 2.6.1, helmikuu 2015.
- [27] A. Nüchter, *Efficient 3D Point Cloud Processing with 3DTK - The 3D Toolkit*, ohjelmapuhe, “3D ARCH - 3D Virtual Reconstruction and Visualization of Complex Architectures, 7th International Workshop on”, Nafplio, Greece, maaliskuu 2017.
- [28] K. Klasing, D. Althoff, D. Wollherr ja M. Buss, “Comparison of surface normal estimation methods for range sensing applications”, teoksessa *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, IEEE, 2009, s. 3206–3211.
- [29] H. Badino, D. Huber, Y. Park ja T. Kanade, “Fast and accurate computation of surface normals from range images”, teoksessa *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, IEEE, 2011, s. 3084–3091.
- [30] J. H. Friedman, J. L. Bentley ja R. A. Finkel, “An algorithm for finding best matches in logarithmic expected time”, *ACM Transactions on Mathematical Software (TOMS)*, vol. 3, nro 3, s. 209–226, 1977.
- [31] M. Muja ja D. G. Lowe, “Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration”, teoksessa *International Conference on Computer Vision Theory and Application (VISSAPP'09)*, INSTICC Press, 2009, s. 331–340.
- [32] J. Elseberg, S. Magnenat, R. Siegwart ja A. Nüchter, “Comparison of nearest-neighbor-search strategies and implementations for efficient shape registration”, *Journal of Software Engineering for Robotics*, vol. 3, nro 1, s. 2–12, helmikuu 2012.
- [33] J. L. Bentley, “Multidimensional binary search trees used for associative searching”, *Communications of the ACM*, vol. 18, nro 9, s. 509–517, 1975.
- [34] P. Sanders, S. Lamm, L. Hübschle-Schneider, E. Schrade ja C. Dachsbacher, “Efficient Random Sampling - Parallel, Vectorized, Cache-Efficient, and Online”, *CoRR*, vol. abs/1610.05141, lokakuu 2016. url: <http://arxiv.org/abs/1610.05141>.
- [35] F. Pomerleau, F. Colas, R. Siegwart ym., “A review of point cloud registration algorithms for mobile robotics”, *Foundations and Trends® in Robotics*, vol. 4, nro 1, s. 1–104, heinäkuu 2015.
- [36] D. Nehab ja P. Shilane, “Stratified point sampling of 3D models”, teoksessa *Proc. Eurographics Symp. on Point-Based Graphics*, M Alexa ja S Rusinkiewicz, toim., 2004, s. 49–56.
- [37] S. Rusinkiewicz ja M. Levoy, “Efficient variants of the ICP algorithm”, teoksessa *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*, IEEE, 2001, s. 145–152.

- [38] S. Sotoodeh, “Outlier detection in laser scanner point clouds”, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, nro 5, s. 297–302, syyskuu 2006.
- [39] S. Papadimitriou, H. Kitagawa, P. B. Gibbons ja C. Faloutsos, “LocI: Fast outlier detection using the local correlation integral”, teoksessa *Data Engineering, 2003. Proceedings. 19th International Conference on*, IEEE, 2003, s. 315–326.
- [40] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald ja W. Stuetzle, “Surface reconstruction from unorganized points”, teoksessa *Computer Graphics*, SIGGRAPH, vol. 26, Chicago: ACM, heinäkuu 1992.
- [41] S. Wold, K. Esbensen ja P. Geladi, “Principal component analysis”, *Chemometrics and intelligent laboratory systems*, vol. 2, nro 1-3, s. 37–52, 1987.
- [42] K.-H. Bae ja D. D. Lichti, “A method for automated registration of unorganised point clouds”, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 63, nro 1, s. 36–54, 2008.
- [43] R. B. Rusu, “Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments”, tohtorinväitöskirja, Technische Universität München, syyskuu 2009.
- [44] G. K. Tam, Z.-Q. Cheng, Y.-K. Lai, F. C. Langbein, Y. Liu, D. Marshall, R. R. Martin, X.-F. Sun ja P. L. Rosin, “Registration of 3D point clouds and meshes: a survey from rigid to nonrigid”, *Visualization and Computer Graphics, IEEE Transactions on*, vol. 19, nro 7, s. 1199–1217, 2013.
- [45] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha ja M. Beetz, “Towards 3D point cloud based object maps for household environments”, *Robotics and Autonomous Systems*, vol. 56, nro 11, s. 927–941, 2008.
- [46] J. Salvi, C. Matabosch, D. Fofi ja J. Forest, “A review of recent range image registration methods with accuracy evaluation”, *Image and Vision computing*, vol. 25, nro 5, s. 578–596, 2007.
- [47] Y. Chen ja G. Medioni, “Object modeling by registration of multiple range images”, teoksessa *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, vol. 3, huhtikuu 1991, s. 2724–2729.
- [48] T. D. Stoyanov, M. Magnusson, H. Andreasson ja A. Lilienthal, “Fast and accurate scan registration through minimization of the distance between compact 3D NDT representations”, *The International Journal of Robotics Research*, vol. 31, nro 12, s. 1377–1393, 2012.
- [49] M. Magnusson, N. Vaskevicius, T. Stoyanov, K. Pathak ja A. Birk, “Beyond points: Evaluating recent 3D scan-matching algorithms”, teoksessa *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, IEEE, 2015, s. 3631–3637.
- [50] P. J. Besl ja N. D. McKay, “Method for registration of 3-D shapes”, *IEEE Transactions on pattern analysis and machine intelligence*, vol. 14, nro 2, s. 239–256, helmikuu 1992.

- [51] A. Segal, D. Haehnel ja S. Thrun, “Generalized-ICP”, teoksessa *Robotics: Science and Systems*, vol. 2, 2009.
- [52] K. Pulli, “Multiview registration for large data sets”, teoksessa *3-D Digital Imaging and Modeling, 1999. Proceedings. Second International Conference on*, IEEE, 1999, s. 160–168.
- [53] G. Blais ja M. D. Levine, “Registering multiview range data to create 3D computer objects”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, nro 8, s. 820–824, 1995.
- [54] T. Zinßer, J. Schmidt ja H. Niemann, “A refined ICP algorithm for robust 3-D correspondence estimation”, teoksessa *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, IEEE, vol. 2, syyskuu 2003, s. 695–698.
- [55] M. Korn, M. Holzkothén ja J. Pauli, “Color supported generalized-ICP”, teoksessa *Computer Vision Theory and Applications (VISAPP), 2014 International Conference on*, IEEE, vol. 3, 2014, s. 592–599.
- [56] G. Turk ja M. Levoy, “Zippered Polygon Meshes from Range Images”, teoksessa *SIGGRAPH '94, Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, New York, NY, USA: ACM, 1994, s. 311–318.
- [57] K. S. Arun, T. S. Huang ja S. D. Blostein, “Least-squares fitting of two 3-D point sets”, *IEEE Transactions on pattern analysis and machine intelligence*, nro 5, s. 698–700, 1987.
- [58] F. Pomerleau, F. Colas, R. Siegwart ja S. Magnenat, “Comparing ICP variants on real-world data sets”, *Autonomous Robots*, vol. 34, nro 3, s. 133–148, 2013.
- [59] D. Holz, A. E. Ichim, F. Tombari, R. B. Rusu ja S. Behnke, “Registration with the Point Cloud Library: A modular framework for aligning in 3-D”, *IEEE Robotics & Automation Magazine*, vol. 22, nro 4, s. 110–124, 2015.
- [60] D. Girardeau-Montaut, *Distances Computation*, CloudCompareWiki, online [25.4.2016], helmikuu 2015. url: [http://www.cloudcompare.org/doc/wiki/index.php?title=Distances\\_Computation](http://www.cloudcompare.org/doc/wiki/index.php?title=Distances_Computation).
- [61] D. Girardeau-Montaut, M. Roux, R. Marc ja G. Thibault, “Change detection on points cloud data acquired with a ground laser scanner”, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, tammikuu 2005.
- [62] *Leica HDS6200 User Manual*, Version 3.0 English, Leica Geosystems, Heinrich-Wild-Strasse, CH-9435 Heerbrugg, Switzerland, 2010.
- [63] *Z+F SDK User Guide*, Zoller + Fröhlich GmbH, Simoniusstrasse, 88239 Wangen im Allgäu, Germany.
- [64] C. A. Brewer, “Color use guidelines for data representation”, teoksessa *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 1999, s. 55–60.